Pairing Inequalities and Stochastic Lot-Sizing Problems: A Study in Integer Programming

A Thesis
Presented to
The Academic Faculty

by

Yongpei Guan

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

School of Industrial & Systems Engineering
Georgia Institute of Technology
August 2005

Pairing Inequalities and Stochastic Lot-Sizing Problems: A Study in Integer Programming

Approved by:

Dr. George L. Nemhauser, Advisor
School of Industrial & Systems Engineering
Georgia Institute of Technology

Dr. Shabbir Ahmed, Co-advisor
School of Industrial & Systems Engineering
Georgia Institute of Technology

Dr. John J. Bartholdi, III
School of Industrial & Systems Engineering
Georgia Institute of Technology

Dr. Zonghao Gu
CPLEX Development Team
ILOG Inc.

Dr. Samer Takriti
Thomas J. Watson Research Center
IBM Inc.

Date Approved: July 15, 2005

To Lei Tian,

and our parents

Xiaohang Hu, Jinlian Huang, Zhongshan Lu and Haiqing Tian.

ACKNOWLEDGEMENTS

I would like to thank my advisors for their encouragement and helpfulness during my time at Georgia Tech. I want to express my appreciation to George L. Nemhauser for his guidance, honest advice, financial support, and for sharing some of his wisdom with me. Many thanks to Shabbir Ahmed for his assistance in developing the ideas in this thesis and for his patient attention to details and to John J. Bartholdi, III for his initial financial support, suggestions regarding my research and many conversations he has taken the time to have with me.

I am also grateful to Zonghao Gu and Samer Takriti for their willingness to be on my dissertation committee, as well as the great industrial experience shared with me. Many thanks to Anita Race and Pam Morrison for their administrative support. Thanks also to Alper Atamtürk, Ismail de Farias, Diego Klabjan, Andrew J. Miller and Craig Tovey for their suggestions and support.

I would like to thank current and former fellow students at Georgia Tech for frequent discussions and for their friendship. Thanks to Vijay Bharadwaj, James Paul Brooks, Brady Hunsaker, Ahmet Keta and Dieter Vandenbussche for their friendship and assistance. Thanks also to Juan Pablo Vielma Centeno, Renan Garcia, Marcos Goycoolea, Kai Huang, Yetkin Ileri, Yun Fong Lim, Zhaosong Lu, Jim Luedtke, Abhyuday Mandal, Jerry O'Neal, Zhiguang Qian and Sriram Subramanian for their friendship and advice.

In addition, I would like to thank all of my friends not associated with my dissertation work for their encouragement and friendship.

Finally, I would like to express thanks to my wife, Lei Tian, for her support, her friendship and her love.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

This thesis capitalizes on recent success in stochastic linear programming and mixed integer programming to develop new solution methods for stochastic integer programming.

We first study a simple and important stochastic integer programming problem, called stochastic uncapacitated lot-sizing (SLS), which is motivated by production planning under uncertainty. We describe a multi-stage stochastic integer programming formulation of the problem and develop a family of valid inequalities, called the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequalities. We establish facet-defining conditions and show that these inequalities are sufficient to describe the convex hull of integral solutions for two-period instances. A separation heuristic for $(\mathcal{Q}, S_{\mathcal{Q}})$ inequalities is developed and incorporated into a branch-and-cut algorithm. A computational study verifies the usefulness of the inequalities as cuts.

Then, motivated by the polyhedral study of $(\mathcal{Q}, S_{\mathcal{Q}})$ inequalities for SLS, we analyze the underlying integer programming scheme for general stochastic integer programming problems. We present a scheme for generating new valid inequalities for mixed integer programs by taking pair-wise combinations of existing valid inequalities. The scheme is in general sequence-dependent and therefore leads to an exponential number of inequalities. For some special cases, we identify combination sequences that lead to a manageable set of all non-dominated inequalities. We also analyze the conditions such that the inequalities generated by our approach are facet-defining and describe the convex hull of integral solutions. We illustrate the framework for some deterministic and stochastic integer programs and present computational results which show the efficiency of adding the new generated inequalities as cuts.

CHAPTER 1

INTRODUCTION

The mathematical programming or optimization approach to model and solve decision problems started in the 1940's, motivated initially by the need to solve complex planning problems in military operations during World War II. Seminal research by such pioneers as George Dantzig and John von Neumann led to the development of the simplex method and duality theory that provide the fundamental framework for linear programming (LP). In the postwar period, mathematical programming developed rapidly as many industries benefited from applying optimization techniques to their business. Applications included production planning, airline scheduling, resource allocation, shipping or telecommunication networks, oil refining, stock and portfolio selection.

LP is a specific class of mathematical programming problem, in which a linear function is minimized (or maximized) subject to linear inequality and equality constraints. The general form of an LP is

$$\begin{aligned} \min \quad & cx \\ Ax &= b \\ x &\geq 0, \end{aligned} \tag{1.1}$$

where $c \in \mathbb{R}^n$, $A \in \mathbb{R}^{m \times n}$, and $b \in \mathbb{R}^m$ are problem parameters and $x \in \mathbb{R}^n_+$ are decision variables.

The *simplex method* developed by Dantzig [30] is an efficient approach to solve linear programs. The algorithm passes from vertex to vertex on the boundary of the feasible polyhedron, repeatedly decreasing the objective function until either an optimal solution is found, or it is established that no optimal solution exists. In principle, the algorithm does not run in polynomial time. In practice, however, the method is highly efficient, typically requiring a number of steps which is just a small multiple of the number of constraints.

The first polynomial algorithm, called the *ellipsoid* method, to solve LP problems was developed by Khachiyan [58] in 1979. The ellipsoid method is faster than the simplex method on some contrived cases where the simplex method performs poorly. In practice, however, the simplex method is much faster than the ellipsoid method. In 1984, Karmarkar [57] introduced a polynomial interior point method for linear programming, combining the desirable theoretical properties of the ellipsoid method and practical advantages of the simplex method. The algorithm does not search from vertex to vertex to find better solution at each step, but passes through the interior of the feasible region. Interior point methods are now generally considered competitive with the simplex method with the exception that the simplex method performs much better for reoptimization, i.e., when an instance needs to be resolved after small changes to the problem.

Sophisticated optimization packages, like CPLEX [54] and XPRESS [36], provide efficient implementations of the simplex method and interior point algorithms. Currently, LP problems involving thousands or even millions of variables can be solved on modern computers.

Detailed information on simplex and interior point algorithms for LP are available in the books by Chvátal [26] and Schrijver [89]. Recent developments and history of LP can be found in the survey paper by Dantzig [33].

The capability of solving huge LPs makes it possible to solve problems that are generalizations of LP. Here, we discuss two generalizations: stochastic programming and integer programming.

1.1 Stochastic Programming

Stochastic programming is a generalization of LP that addresses parameter uncertainty. It is motivated by the fact that uncertainty is a key ingredient in decision problems. Examples of uncertainty include demand uncertainty in each time period in production planning, flight time uncertainty in airline scheduling and stock price uncertainty in financial planning.

1.1.1 Two-Stage Stochastic Linear Programming

Pioneering work in stochastic programming was done by Dantzig [31] and Beale [13] in 1955. Both investigated a classical stochastic optimization problem called two-stage stochastic linear programming with recourse. In this type of mathematical model, random problem parameters are used to represent uncertainty and the objective is to minimize the total expected cost. The general formulation of the two-stage stochastic linear program is

$$\begin{aligned} z = \min \quad cx + E_\omega Q(x,\omega) & \\ Ax &= b \\ x &\geq 0, \end{aligned} \tag{1.2}$$

where

$$\begin{aligned} Q(x,\omega) = \min \quad f(\omega)y & \\ D(\omega)y &= d(\omega) + B(\omega)x \\ y &\geq 0. \end{aligned} \tag{1.3}$$

The above formulation (1.2) and (1.3) is among the simplest possible dynamic decision process given by the following two-stage scheme

$$\text{Decision on } x \;\rightarrow\; \text{Observation on } \omega \;\rightarrow\; \text{Decision on } y.$$

Here the decision variable has been split into two parts, $x \in \mathbb{R}^{n_1}$ and $y \in \mathbb{R}^{n_2}$, with the convention that y is allowed to depend on ω, where ω is an element of the probability space and represents an outcome of random parameters. Formulation (1.2) is considered the *first-stage* problem and formulation (1.3) is called the *second-stage* problem. The matrix A and the vector b are known with certainty, which corresponds to the second stage uncertain information. The vector y is referred to as *recourse* decisions to reflect the idea that the components of y are chosen to compensate for random events. That is, once x is fixed and ω is observed, the resulting second-stage problem is deterministic. Correspondingly, the function $Q(x,\omega)$ is referred to as the *recourse function* and E_ω denotes the expectation with respect to ω. The technology matrix $D(\omega)$, the right-hand side $d(\omega)$, the transition matrix $B(\omega)$, and the objective function coefficients $f(\omega)$ of this linear program can be random. The case where (1.3) is feasible for any choice of x and ω is called complete recourse since there is a feasible recourse decision y for any first stage decision x and uncertainty ω. The recourse function $Q(x, \omega)$ here is a nonsmooth convex function. Wets [95] gave a detailed analysis of properties of complete recourse. The complete recourse problem involves linear constraints and a convex objective function. Thus the developments in linear programming and nonlinear programming in the 1940's made it possible to study stochastic linear programming.

By using a finite set of scenarios of ω, formulation (1.2) and (1.3) generates a deterministic equivalent formulation. Algorithms for solving these formulations were presented by Wets [96, 97]. More information about the early work on algorithmic approaches can be found in the book by Kall [56].

Decomposition methods are important in solving large-scale two stage stochastic linear programs. In the early 1960's, Dantzig and Madansky [34] provided an idea to solve two-stage stochastic linear programs by using Dantzig-Wolfe decomposition [35] algorithms to solve the dual. Soon after that, the Benders decomposition algorithm [15] was developed in 1962, and later in 1969, Van Slyke and Wets [84] provided their L-shaped method to solve two-stage stochastic linear programs. The basic idea of the L-shaped method is similar to Benders decomposition. Expected-value cuts, which represents an outer linearization of the recourse function, is used. Birge and Louveaux [18] provided a variant of the L-shaped method by using multiple cuts. In their approach, one additional different cut with respect to each scenario is computed.

Several nonlinear programming methods were also proposed. Stochastic quasigradient methods were originally studied by Ermoliev [38, 39, 40], who provided the theory and algorithms. However, the convergence rates of stochastic quasigradient methods are slow. The performance of the algorithms depends on how to specify objective values, subgradients and step-sizes. Later on, the stepsize rules and stopping criteria for stochastic quasigradient methods were provided by Pflug [80]. Ruszczyński [87] proposed a linearization method and Marti [74] introduced a semistochastic approximation approach to improve the convergence of stochastic quasigradient methods. Besides stochastic quasigradient methods, Ruszczyński [86] described a regularized decomposition method for minimizing a sum of polyhedral functions, which can be directly applied to solve stochastic linear programs. Trust region methods for stochastic programming were implemented by Linderoth and Wright [66]. Interior-point linear programming methods on two-stage stochastic programs were studied by Lustig *et al.* [72]. Higle and Sen [52, 51] proposed a stochastic decomposition Benders-type decomposition method. It is similar to the stochastic quasigradient algorithm and asymptotically creates an outer linearization of the second-stage costs. Later on, Higle *et al.* [50] extended the framework to a multi-cut formulation.

1.1.2 Multi-Stage Stochastic Linear Programming

Two-stage stochastic linear programming considers the entire uncertain future as the second stage. However, most practical problems involve a sequence of decisions that react to outcomes evolving over time. The basic scheme of alternating observations and decisions has the form

$$\ldots \;\rightarrow \text{Decision on } x_{t-1} \;\rightarrow\; \text{Observation on } \omega_t \;\rightarrow\; \text{Decision on } x_t \;\rightarrow \ldots.$$

The process extends over $t = 1, \ldots, T$ stages where $x_t \in \mathbb{R}^{n_t}$ represents the decision to be made in the tth stage and $\omega_t \in \mathbb{R}^k_t$ represents a random variable revealed or known at the beginning of stage t.

The stochastic program corresponding to this multistage decision problem leads to a deterministic equivalent multi-stage stochastic linear program with a special structure. This structure can be interpreted as a scenario tree with T levels (or stages) as shown in Figure 1, where a node i in stage t of the tree gives the state of the system that can be distinguished by information available up to stage t. Each node i of the scenario tree, except the root node (indexed as $i = 0$), has a unique parent $a(i)$, and each non-terminal node i is the root of a subtree $\mathcal{T}(i) = (\mathcal{V}(i), \mathcal{E}(i))$ where $\mathcal{V}(i)$ and $\mathcal{E}(i)$ represent the set of nodes and arcs in the subtree respectively. For notational brevity we use $\mathcal{T} = \mathcal{T}(0)$ and $\mathcal{V} = \mathcal{V}(0)$ for the whole tree. The set of leaf nodes of $\mathcal{T}$ is denoted by $\mathcal{L}$. The probability associated

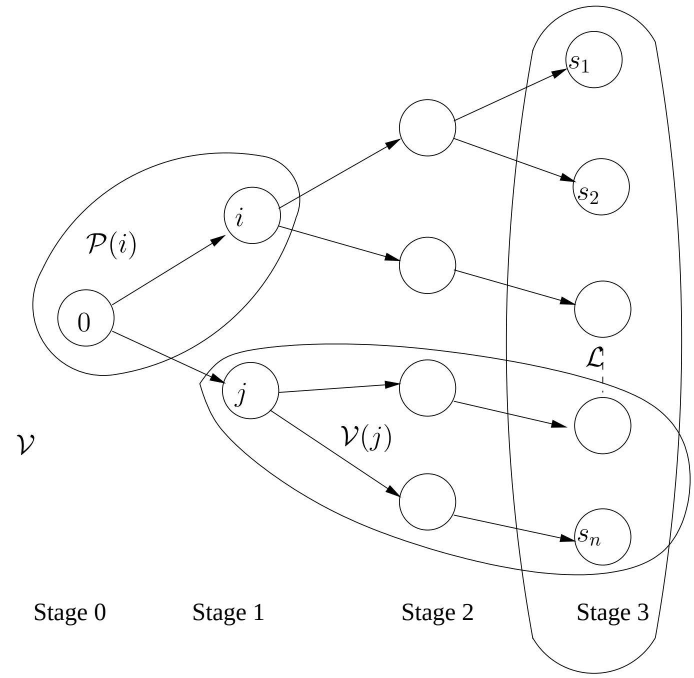

Figure 1: Multi-stage stochastic scenario tree formulation

with the state represented by node i is p_i. The set of nodes on the path from the root node to node i is denoted by $\mathcal{P}(i)$. If $i \in \mathcal{L}$ then $\mathcal{P}(i)$ corresponds to a *scenario*, and represents a joint realization of the problems parameters over all periods $1, \ldots, T$. We define $\mathcal{P}(i,j) = \{k : k \in \mathcal{P}(j) \cap \mathcal{V}(i)\}$, thus $\mathcal{P}(i) = \mathcal{P}(0,i)$. We let $\mathcal{C}(i)$ denote the set of nodes that are immediate children of node i, i.e., $\mathcal{C}(i) = \{j : a(j) = i\}$; $t(i)$ denote the time stage or level of node i in the tree, i.e., $t(i) = |\mathcal{P}(i)|$; and $\mathcal{L}(i)$ denote the leaf nodes of the subtree $\mathcal{T}(i)$.

The multi-stage stochastic extension of a deterministic linear program for a T period problem with the objective of minimizing total expected cost can be formulated as follows:

$$\begin{aligned}
\min \ Z = \quad & \textstyle\sum_{i \in \mathcal{V}} p_i c_i x_i & & \\
& A_1 x_1 & &= b_1 \\
& \textstyle\sum_{m \in \mathcal{P}(a(i))} B_m x_m + A_i x_i & &= b_i, \ \forall i \in \mathcal{V} \\
& x_i \geq 0, & & \quad \ \ \forall i \in \mathcal{V}
\end{aligned}$$

If a multi-stage stochastic linear program has block-separable recourse, the problem is easier to solve. Louveaux [70] studied multi-stage stochastic linear programming problems with this type of structure. Beale *et al.* [14] studied a first-order approach to a class of multi-stage stochastic programming problems. Other methods are mainly based on the developed L-shaped method for two-stage stochastic linear programs. Birge [17] provided decomposition and partitioning methods for multi-stage stochastic linear programming. He extended the L-shaped method for the two-stage problem to the multi-stage setting by exploying a nested Benders decomposition scheme. Gassmann [43] explored different tree-traversing strategies in a Benders decomposition framework for stochastic multi-stage programs, which is based on results developed by Wittrock [99] for deterministic multi-stage programs. Pereira and Pinto [79] presented stochastic dual dynamic programming that exploits the piecewise linear property of the recourse function and introduced path sampling for obtaining estimates of upper bounds.

With significant development of computer power in the 1980's, parallel computing presents an efficient way to solve large-scale stochastic linear programs as suggested by Dantzig [32] and Wets [98]. Other developments in parallel computing can be found in Zenios [101], Hillier and Eckstein [53], and Ariyawansa and Hudson [7].

1.2 Mixed Integer Programming

Mixed integer programming (MIP) is another generalization of LP. Integer variables are indispensable in the mathematical formulations of practical problems. In these models, integrality may be required due to indivisible or boolean decisions. Introducing integer variables is also required when modelling logical relationships and handling discontinuous and piecewise linear functions. In this section, we describe some technical background needed to describe MIP algorithms. Comprehensive references for MIP are the books by Nemhauser and Wolsey [76], Wolsey [100], and Schrijver [89]. Much of the material we present in this section can be found in any of these books.

The general formulation of a MIP is

$$\begin{aligned} \min \quad & c_1 x + c_2 y \\ & Ax + By \le b \\ & x \ge 0,\ y \ge 0 \quad \text{integer}, \end{aligned} \tag{1.4}$$

where $c_1 \in \mathbb{R}^{n_1}, c_2 \in \mathbb{R}^{n_2}, b \in \mathbb{R}^m, A \in \mathbb{R}^{m\times n_1}$ and $B \in \mathbb{R}^{m\times n_2}$ are problem parameters and $x \in \mathbb{R}^{n_1}_+$ and $y \in \mathbb{Z}^{n_2}_+$ are decision variables. The LP *relaxation* of MIP is obtained by relaxing the requirement $y \in \mathbb{Z}^{n_2}$ to $y \in \mathbb{R}^{n_2}$. The optimal objective value of the LP relaxation is a lower bound on the optimal objective value of MIP. We denote the feasible region for the MIP as X_{MIP} and the feasible region of the LP relaxation for the MIP as X_{P}.

When there are no continuous variables, we have the special case of an integer program (IP). MIP and IP are in general harder to solve than LP. Khachiyan [58] showed that LP is polynomially solvable while Cook [27] showed that the general IP is $\mathcal{NP}$-hard, which means that it is unlikely that a polynomial time algorithm exists for solving it. Some problems such as the assignment, matching and shortest path problems are polynomially solvable. Other problems such as the travelling salesman problem and capacitated lot-sizing problem are $\mathcal{NP}$-hard. In the last 10-20 years, great progress has been made in our ability to solve MIPs using an algorithmic framework known as *branch-and-cut*, which combines *branch-and-bound* with *cutting planes*. We now describe this approach.

1.2.1 Branch-and-Bound

LP based branch-and-bound, introduced by Land and Doig [62], is a traditional method to solve integer programs. It is an implicit enumeration method that uses a search tree to find an optimal solution for (1.4). The basic idea of this method is to control the enumeration of integer feasible solutions by using LP relaxations to conclude that certain parts of the tree cannot give a better solution than what is already known. It starts from the LP relaxation and sets the lower bound LB $= -\infty$ and upper bound UB $= +\infty$, respectively. If the initial solution (x^*, y^*) of the LP relaxation of (1.4) has y^* integer, then an optimal integer solution is obtained. If not, there exists an element y^*_j which is fractional and we can branch

y_j^* and create two different IP subproblems, also called the two child nodes in the tree. One has the additional constraint $y_j \geq \lceil y_j^* \rceil$, the other $y_j \leq \lfloor y_j^* \rfloor$, where "$\lceil \ \rceil$" and "$\lfloor \ \rfloor$" are the round-up and round-down of y_j^* respectively. For each node, we solve the LP relaxation of the MIP corresponding to this node and do the following operations.

1. If the LP relaxation solution is integral, we have found a feasible integer solution. If this solution has a smaller objective value than the current upper bound, we update the upper bound to be this objective value. The current node is fathomed.

2. If the LP relaxation is infeasible, then the corresponding MIP is also infeasible and no further branching is required. The current node is fathomed.

3. If the objective value of the LP relaxation is larger than current upper bound, then any descendent of this node cannot yield a better integer solution and hence no further branching is required. The current node is fathomed.

4. Otherwise, we call this current node "active," branch this node in the tree and continue the branch-and-bound process. The current node is fathomed.

The algorithm terminates when all nodes are *fathomed*. At this point, the algorithm has found an optimal solution or has shown that the original problem is infeasible. It is also possible to modify the algorithm so that it terminates within a specified tolerance, i.e., when $(\mathrm{UB} - \mathrm{LB})/\mathrm{LB} < \alpha$ for some $\alpha > 0$.

In branch-and-bound, tight upper bounds and lower bounds are important in reducing the size of the search tree and therefore speeding the algorithm significantly.

> *Upper bound:* The objective value corresponding to any feasible integer solution provides an upper bound for the problem.
>
> *Lower bound:* The optimal objective value of any LP relaxation for the MIP subproblem corresponding to any node provides a lower bound for this node. The smallest lower bound among all *active* nodes will be the lower bound of the entire problem.

Making the upper bound smaller will speed up the algorithm. Since the objective value corresponding to any feasible integer solution provides an upper bound, heuristic methods can be developed to improve the upper bound.

Making the lower bound larger will also speed up the algorithm. *Cutting plane algorithms* that we describe in Section 1.2.2 can be used to improve the lower bounds.

1.2.2 Cutting Planes

A *valid inequality* $\pi^1 x + \pi^2 y \leq \pi_0$ for X_{MIP} is an inequality that is satisfied by all $(x, y) \in X_{\text{MIP}}$. A *cut* or *cutting plane* is a valid inequality that excludes some fractional points for X_{P}. The *convex hull* of a set $S = \{(x, y) : Ax + by \leq b, x \geq 0, y \geq 0 \text{ integer}\}$, denoted by $\text{conv}(S)$, is the set of points that can be represented as convex combinations of points in S. An important theoretical result of integer programming is that such a set is *polyhedral*. This means that it can be represented by a finite set of linear inequalities. If we have the full convex hull description of the feasible integer solutions defined in (1.4), then the problem can be solved as an LP instead of a MIP. Normally, the full convex hull description requires a number of inequalities exponential in the size of the input. However, for practical problems, we may only need to add a few valid inequalities to obtain an optimal integral solution. Specifically, we need valid inequalities that are violated by optimal solutions to the LP relaxations. After adding these inequalities as cuts, the lower bound for each node in the enumeration tree may increase significantly. In the branch-and-bound algorithm, if we add these inequalities to the LP relaxation only at the root node, the algorithm is called a *cut-and-branch* algorithm. Otherwise, if these inequalities are embedded throughout the enumeration tree, the algorithm is referred as a *branch-and-cut* algorithm. Suppose, we solve the LP relaxation of a subproblem and obtain a nonintegral solution y^*. Then, we would like to find a cut or the most violated cut $\pi y \leq \pi_0$ such that $\pi y^* \geq \pi_0$. This search process for such inequalities is referred to as the *separation problem*. Therefore, a family of strong cuts that cut off a significant part of the LP feasible region and efficient *separation algorithms* are important to improve the branch-and-cut algorithm.

As mentioned in Johnson, Nemhauser and Savelsbergh [55], typically there are three types of valid inequalities that can be used in a branch-and-cut algorithm. The first type of cuts does not have any particular structure as the cuts are based only on variables being integral or binary. Such cuts have the advantage of being general at the cost of not being strong. Chvátal-Gomory (CG) inequalities are examples of these cuts. They were introduced by Gomory [44] in the 1950's and given a different interpretation by Chvátal [24]. For a given inequality

$$\sum_j a_{ij} y_j \leq b_i, y_j \geq 0 \text{ integer}, \tag{1.5}$$

a CG cut is given by

$$\sum_j \lfloor a_{ij} \rfloor y_j \leq \lfloor b_i \rfloor \tag{1.6}$$

and is satisfied by all integer solutions that satisfy (1.5). Note that if the a_{ij}s are integers and b_i is not an integer, then (1.6) cuts off fractional points. CG cuts are general and can be derived immediately from a fractional basic solution to the LP relaxation. They are also easy to extend to MIPs.

The second type of cuts are derived from relaxations of the problem. For instance, consider a single row of the constraint set of the form

$$\sum_{j \in N} a_j y_j \leq b, y_j \in \{0, 1\} \text{ for } j \in N.$$

We can assume $a_j > 0$ since we can replace $y_j = 1 - y_j$ if $a_j < 0$. Let C be a minimal set such that

$$\sum_{j \in C} a_j > b.$$

Then C is called a *minimal cover* and the corresponding *cover inequality* [29] is

$$\sum_{j \in C} y_j \leq |C| - 1.$$

Cover inequalities can be strengthened by a process known as *lifting* to include the variables $y_j, j \notin C$.

The third type of cuts are problem-specific. Examples are *sub-tour elimination* inequalities for the travelling salesman problem given by Chvátal [25] and (ℓ, S) *inequalities* for lot-sizing problems that we will describe in Section 1.4.

In the remainder of this section, we describe some general purpose cuts for generating new valid inequalities based on several existing valid inequalities. These cuts are related to the procedure described in Chapter 5.

Balas [10] described a general disjunctive approach for obtaining valid inequalities. Given two inequalities $\sum_{j \in N} \pi_j^1 y_j \leq \pi_0^1$ and $\sum_{j \in N} \pi_j^2 y_j \leq \pi_0^2$ that are valid for $S_1 \subset R_+^n$ and $S_2 \subset R_+^n$ respectively, then the *disjunctive inequality*

$$\sum_{j \in N} \min(\pi_j^1, \pi_j^2) y_j \leq \max(\pi_0^1, \pi_0^2)$$

is valid for $S_1 \cup S_2$.

The *split cuts* of Cook, Kannan and Schrijver [28] are derived via a disjunctive argument. Corresponding to formulation 1.4, given any $\pi \in Z^{n_2}$ and $\pi_0 \in Z$, each feasible solution must satisfy either $\pi y \leq \pi_0$ or $\pi y \geq \pi_0 + 1$. Then, if we define $P_1 := \{(x, y) \in X_{\mathrm{P}} : \pi y \leq \pi_0\}$ and $P_2 := \{(x, y) \in X_{\mathrm{P}} : \pi y \geq \pi_0 + 1\}$, we have that any inequality which is valid for both P_1 and P_2 is also valid for X_{MIP}. Any inequality of this type is a split cut.

Günlük and Pochet [48] investigated a general mixing procedure to generate two new valid inequalities by combining known valid inequalities. Given a set of inequalities, they first generated a Mixed Integer Rounding inequality (MIR) [77] for each inequality. Then they generate two valid inequalities based on these MIR inequalities.

The approaches described above are general and provide valid inequalities. We explore this further in Chapter 5.

1.3 Stochastic Integer Programming

As in the deterministic case, integer variables are necessary to model applied problems that contains uncertainty. However, stochastic MIP inherits both the complexity of MIP and the complexity of stochastic LP. Therefore, until the significant development of computer power in the 1980's, the systematic investigation of stochastic integer programming algorithms was not possible. Klein Haneveld and van der Vlerk [60] surveyed general stochastic integer programming models and algorithms. Römisch and Schultz [85] provided an introduction to multistage stochastic integer programs. There is a chapter in the book by Birge and

Louveaux [19] and a chapter written by Louveaux and Schultz in the handbook edited by Ruszczyński and Shapiro [88] describing the basic modelling and algorithms for two-stage and multi-stage stochastic integer programs.

The first step to study stochastic integer programs is to consider two-stage problems where some or all of the recourse variables are integer. Stochastic programs with a simple integer recourse variable were first studied by Louveaux and van der Vlerk [69]. Later on, Klein Haneveld and van der Vlerk [59] provided an algorithm to construct convex hulls for this type of problem. Currently, almost all work on stochastic integer programming problems focus on decomposition methods.

1.3.1 Decomposition Methods

The first use of decomposition methods in stochastic integer programs was the integer L-shaped method proposed by Laporte and Louveaux [63] for the case of first-stage binary variables. Laporte *et al.* [64] applied the integer L-shaped algorithm for the capacitated vehicle routing problem with stochastic demands. The generalization of the integer L-Shaped method to mixed integer first-stage stochastic integer programs was studied in the dissertation by Carøe [20] and in the papers by Carøe and Tind [22, 23].

Ahmed *et al.* [5] provided the first attempt to design a method based on branching first-stage continuous variables. Solution approaches based on enumeration and bounding in the first-stage and handling the second-stage by exploring the similarities using algebraic methods was presented by Schultz *et al.* [90].

Little structure was explored for multi-stage stochastic integer programs. Römisch and Schultz [85] provided an introduction for this type of problem. Many practical problems have multi-stage stochastic integer structure. For instance, Dentcheva and Römisch [37] studied optimal power generation under uncertainty, Nowak and Römisch [78] provided stochastic Lagrangian relaxation applied to power scheduling and Takriti *et al.* [92] studied the unit commitment problem. Recently, Ahmed *et al.* [3, 4] studied a stochastic capacity expansion problem. Alonso-Ayuso *et al.* [6] provided a branch-and-fix coordination approach to solving multi-stage stochastic integer programs. They formulated the scenario based multi-stage formulation plus non-anticipativity constraints. A scenario-wise LP-based branch-and-bound method was used in which non-anticipativity constraints are satisfied by a coordinated fixing of variables that puts variables for different scenarios at identical values.

1.3.2 Polyhedral Results

There is little research on polyhedral results in stochastic integer programming, especially multi-stage stochastic integer programming problems. Carøe and Schultz [21] provided a dual decomposition method for stochastic integer programming. Sen and Sherali [91] developed a decomposition method with branch-and-cut for two-stage stochastic mixed-integer programming. All of these algorithms combine cutting planes within a decomposition algorithm to approximate the second-stage value function.

In this dissertation, we perform polyhedral studies based on the deterministic equivalent formulation. We use lot-sizing as an important instance to begin with, and then we extend the idea to general stochastic integer programming problems.

1.4 Lot-Sizing Problem

The fundamental deterministic uncapacitated lot-sizing problem is to determine a minimum cost production and inventory holding schedule for a product so as to satisfy its demand over a finite discrete-time planning horizon. A standard mixed-integer programming formulation for the single item, uncapacitated, lot-sizing problem is (cf. [76]):

$$
\begin{aligned}
\text{(LS)}: \quad \min \quad & \sum_{i=0}^{T} (\alpha_i x_i + \beta_i y_i + h_i s_i) \\
& s_{i-1} + x_i = d_i + s_i \qquad i = 0, \ldots, T, \\
& x_i \leq M_i y_i \qquad i = 0, \ldots, T, \\
& x_i, s_i \geq 0, \ y_i \in \{0, 1\} \quad i = 0, \ldots, T,
\end{aligned}
$$

where x_i represents the production in period i, s_i represents the inventory at the end of period i, and y_i indicates if there is a production set-up in period i. Problem parameters α_i, β_i, h_i, and d_i represent the production cost, set-up cost, holding cost, and the demand in period i, respectively. Since there is no restriction on the production level, the parameter

M_i is a sufficiently large upper bound on x_i. This bound can be set as $M_i = \sum_{j=i}^{T} d_j$. We denote the set of feasible solutions of (LS) as X_{LS}.

Although (LS) is solvable in strongly polynomial time using specialized dynamic programming algorithms (cf. [1, 41, 93, 94]), such algorithms are not applicable when (LS) is embedded, as it frequently is, in various multi-period production planning problems. This has motivated the polyhedral study of X_{LS} in order to improve integer programming approaches for such production planning problems. Barany *et al.* [11, 12] prove that a complete polyhedral description of the convex hull of X_{LS} is given by original inequalities $x_i \geq 0$ and $0 \leq y_i \leq 1$ together with the (ℓ, S) inequalities

$$\sum_{i \in S} x_i + \sum_{i \in \overline{S}} d_{i\ell} y_i \geq d_{0\ell}, \tag{1.7}$$

where $\ell \in \{0, 1, \ldots, T\}$, $S \subseteq \{0, 1, \ldots, \ell\}$, $\overline{S} = \{0, 1, \ldots, \ell\} \setminus S$, and $d_{ij} = \sum_{k=i}^{j} d_k$. The authors reported good computational results for multiple item capacitated lot-sizing problems using the (ℓ, S) inequalities within a branch-and-cut scheme. Krarup and Bilde [61] present a reformulation by introducing new variables q_{ij} representing the quantity produced in time period i to satisfy the demand in time period j for all $j \in \{i, \ldots, T\}$. Both approaches provide the same tight LP lower bound equal to the optimal objective value of the corresponding integer formulations. Following Barany *et al.*'s work, polyhedral structures of variants of (LS) have been investigated. These include variants of (LS) involving sales and safety stocks [68], start-up costs [83], piecewise linear and concave production costs [2], and constant [65, 82], as well as dynamic [75, 81] production capacities, only to name a few.

1.5 Thesis Outline

In Chapter 2, we introduce the stochastic uncapacitated lot-sizing problem and show that the classical (ℓ, S) inequalities for the deterministic lot-sizing polytope are also valid for the stochastic lot-sizing polytope. Then, we study two formulations of the stochastic uncapacitated lot-sizing problem extended from the traditional deterministic setting. We first show that neither of them can guarantee integral solutions. Then, we prove that these two formulations provide the same tight linear programming lower bound. This conclusion can be extended to general production planning problems with embedded stochastic lot-sizing substructures.

We begin by studying the polyhedral properties for the stochastic uncapacitated lot-sizing problem in Chapter 3, where we develop valid inequalities necessary to describe the convex hull of the two-period stochastic uncapacitated lot-sizing problem. Then, in Chapter 4, we extend the (ℓ, S) inequalities to a general class of valid inequalities, called the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequalities. We establish necessary and sufficient conditions which guarantee that the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequalities are facet-defining. We develop a separation heuristic for the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequalities and then incorporate the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequalities into a branch-and-cut algorithm. A computational study verifies the usefulness of the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequalities as cuts.

In the second half of the dissertation, we provide an underlying theme to study general stochastic integer programs that can be represented by a finite set of scenarios. In Chapter 5, motivated by the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequalities for the stochastic uncapacitated lot-sizing problem, we develop a scheme for generating new valid inequalities for mixed integer programs by taking pair-wise combinations of existing valid inequalities. The scheme is in general sequence-dependent and therefore leads to an exponential number of inequalities. For some important cases, we identify combination sequences that lead to a manageable set of non-dominated inequalities. We illustrate the framework for several deterministic and stochastic integer programs and present computational results which show the efficiency of adding the new generated inequalities as cuts.

Chapter 6 describes the extension of pair-wise combination to the general stochastic scenario tree setting, which is a fundamental structure for stochastic integer programming problems. The scheme is in general sequence-dependent. For this stochastic scenario tree case, we identify combination sequences that lead to non-dominated inequalities. We give sufficient conditions for inequalities generated by our approach to be facet-defining and provide the convex hull. We also introduce separation algorithms. Then, we demonstrate our general approach for generating valid inequalities for different stochastic lot-sizing problems. We also show that for two period cases of these stochastic lot-sizing problems, the valid inequalities generated by our approach describe the convex hull of integral solutions.

Computational experiments show the efficiency of adding these inequalities as cuts for multi-item stochastic capacitated lot-sizing problems.

Finally in Chapter 7, we provide general conclusions and suggestions for future research.

CHAPTER 2

TWO FORMULATIONS OF THE STOCHASTIC UNCAPACITATED LOT-SIZING PROBLEM

2.1 Introduction

In Section 1.4, we described the deterministic lot-sizing problems. In this chapter, we study the stochastic extensions of the deterministic uncapacitated lot-sizing problem. As discussed in Chapter 1, the deterministic uncapacitated lot-sizing problem is to determine the production level so that demand is satisfied at a minimal cost. The lot-sizing model (LS) assumes that the cost and demand parameters are known with certainty for all periods of the planning horizon. However, in many applications, these parameters are uncertain, and, at best, only some distributional information may be available. In this case, LS can be extended to explicitly address uncertainty by adopting a stochastic programming approach. Haugen *et al.* [49] propose a heuristic strategy for such stochastic lot-sizing problems. Ahmed *et al.* [3] propose an extended reformulation of the uncapacitated stochastic lot-sizing problem of which the LP relaxation is significantly tighter than the standard formulation. They also point out that the Wagner-Whitin optimality condition for deterministic uncapacitated lot-sizing problems, i.e., the principle of "No production is undertaken if inventory is available," does not hold in the stochastic case. The stochastic lot-sizing problem has also been considered as an embedded subproblems in some classes of stochastic capacity expansion problems [4], stochastic batch-sizing problems [71], and stochastic production planning problems [16].

In this chapter, we first describe an original stochastic uncapacitated lot-sizing formulation in Section 2.2. Then, we show that the (ℓ, S) inequalities are also valid for the stochastic uncapacitated lot-sizing problem in Section 2.3. Section 2.4 describes a reformulation by introducing auxiliary decision variables. Finally in Section 2.5, we demonstrate that the two formulations provide the same tight linear program lower bound. The results of this section also appear in the paper by Guan *et al.* [45].

2.2 The Stochastic Uncapacitated Lot-Sizing Problem Formulation

In order to describe the stochastic model, we use the scenario tree notation described in Section 1.1.2. Let problem parameters α_i, β_i, h_i and d_i represent production, set up, inventory holding costs and demand in period $t(i)$ corresponding to the state defined by node i. Let p_i represent the probability associated with the state represented by node i. Without loss of generality, we assume $\alpha_i, \beta_i, h_i, d_i \geq 0$ for each $i \in \mathcal{V}$.

Define decision variable x_i to represent the production in period $t(i)$ corresponding to the state defined by node i. Similarly define s_i be the inventory at the end of period $t(i)$ and y_i be the indicator variable for a production set-up in period $t(i)$ corresponding to the state defined by node i.

Then, a multi-stage stochastic integer programming formulation of the single-item, uncapacitated, stochastic lot-sizing problem is:

$$
\begin{aligned}
(\text{SLS1}): \quad \min \quad & \sum_{i \in \mathcal{V}} p_i(\alpha_i x_i + \beta_i y_i + h_i s_i) \\
& s_{a(i)} + x_i = d_i + s_i \qquad i \in \mathcal{V}, \\
& x_i \leq M_i y_i \qquad\qquad\quad i \in \mathcal{V}, \\
& x_i, s_i \geq 0,\ y_i \in \{0,1\} \quad i \in \mathcal{V},
\end{aligned}
$$

where an upper bound on x_i is given by

$$M_i = \max_{j \in \mathcal{L}(i)} d_{ij}.$$

Upon eliminating variables s_i from SLS1, let $d_{i\ell} = \sum_{j \in \mathcal{P}(i,\ell)} d_j$ and we obtain the reformulation:

$$
\begin{aligned}
\text{(SLS)}: \quad \min \quad & \textstyle\sum_{i \in \mathcal{V}} (\bar{\alpha}_i x_i + \bar{\beta}_i y_i) \\
& \textstyle\sum_{j \in \mathcal{P}(i)} x_j \geq d_{0i} \quad i \in \mathcal{V} && (2.1) \\
& 0 \leq x_i \leq M_i y_i \quad i \in \mathcal{V} && (2.2) \\
& y_i \in \{0, 1\} \quad i \in \mathcal{V} && (2.3)
\end{aligned}
$$

where $\bar{\alpha}_i = p_i \alpha_i + \sum_{j \in \mathcal{V}(i)} p_j h_j$ and $\bar{\beta}_i = p_i \beta_i$. Let X_{SLS} be the feasible region of SLS.

2.3 The (ℓ, S) Inequalities

In this section, we show that the (ℓ, S) inequalities (1.7), for the deterministic lot-sizing problem, are valid for SLS. These inequalities are based on a sequence of consecutive time periods that can be thought of as a path in the scenario tree.

Theorem 2.1 *Given $\ell \in \mathcal{V}$ and $S \subseteq \mathcal{P}(\ell)$, the (ℓ, S) inequality*

$$\sum_{i \in S} x_i + \sum_{i \in \overline{S}} d_{i\ell} y_i \geq d_{0\ell}, \tag{2.4}$$

where $\overline{S} = \mathcal{P}(\ell) \setminus S$, is valid for X_{SLS}.

Proof: The proof is analogous to that of the deterministic case (cf. [11]). Given a point $(x, y) \in X_{\text{SLS}}$, we consider two cases: (a) there exists $i \in \overline{S}$ such that $y_i = 1$, and (b) $y_i = 0$ for all $i \in \overline{S}$.

Case (a): Let $k = \operatorname{argmin}\{t(i) : i \in \overline{S},\ y_i = 1\}$. Then $y_i = 0$ and $x_i = 0$ for all $i \in \overline{S} \cap \mathcal{P}(a(k))$. Hence

$$\sum_{i \in S} x_i + \sum_{i \in \overline{S}} d_{i\ell} y_i \geq \sum_{i \in \mathcal{P}(a(k))} x_i + d_{k\ell} \geq d_{0a(k)} + d_{k\ell} = d_{0\ell}.$$

Case (b): If $y_i = 0$ for all $i \in \overline{S}$, then

$$\sum_{i \in S} x_i + \sum_{i \in \overline{S}} d_{i\ell} y_i = \sum_{i \in \mathcal{P}(\ell)} x_i \geq d_{0\ell}.$$

$\square$

Remark 2.1 *Inequality (2.1) is also an (ℓ, S) inequality with $\overline{S} = \emptyset$.*

2.4 The Stochastic Uncapacitated Lot-Sizing Problem Reformulation

Following the idea of Krarup and Bilde [61] for the deterministic case, let q_{ij} represent the quantity produced in node i to satisfy the demand in node j for all $j \in \mathcal{V}(i)$. By eliminating inventory variables, we obtain a reformulation of SLS as developed in Ahmed *et al.* [3]:

$$\text{(RSLS)}: \quad \min \quad \textstyle\sum_{i \in \mathcal{V}} (\bar{\alpha}_i x_i + \bar{\beta}_i y_i)$$

$$x_i \geq \textstyle\sum_{k \in \mathcal{P}(i,j)} q_{ik} \qquad i \in \mathcal{V}, j \in \mathcal{L}(i) \tag{2.5}$$

$$\textstyle\sum_{j \in \mathcal{P}(i)} q_{ji} = d_i \qquad i \in \mathcal{V} \tag{2.6}$$

$$q_{ij} \leq d_j y_i \qquad i \in \mathcal{V}, j \in \mathcal{V}(i) \tag{2.7}$$

$$0 \leq x_i \leq M_i y_i \qquad i \in \mathcal{V} \tag{2.8}$$

$$q_{ij} \geq 0, \qquad i \in \mathcal{V}, j \in \mathcal{V}(i) \tag{2.9}$$

$$y_i \in \{0, 1\} \qquad i \in \mathcal{V}. \tag{2.10}$$

Let X_{SLSL} be the feasible region of (x, y) satisfying constraints (2.1), (2.2) and (2.4) with $0 \leq y_i \leq 1$ and X_{RSLS} be the feasible region of (x, y, q) satisfying constraints (2.5)–(2.9) with $0 \leq y_i \leq 1$.

2.5 Formulation Comparison

Let $\text{proj}_{\{\text{x,y}\}}(\text{X}_{\text{RSLS}})$ be the projection of X_{RSLS} onto (x, y) space. In this section, we show that $\text{proj}_{\{\text{x,y}\}}(\text{X}_{\text{RSLS}}) = \text{X}_{\text{SLSL}}$ by proving $\text{proj}_{\{\text{x,y}\}}(\text{X}_{\text{RSLS}}) \subseteq \text{X}_{\text{SLSL}}$ and $X_{\text{SLSL}} \subseteq \text{proj}_{\{\text{x,y}\}}(\text{X}_{\text{RSLS}})$ respectively.

Proposition 2.1 $\text{proj}_{\{\text{x,y}\}}(\text{X}_{\text{RSLS}}) \subseteq \text{X}_{\text{SLSL}}$.

Proof: For any $(x, y) \in \text{proj}_{\{\text{x,y}\}}(\text{X}_{\text{RSLS}})$, we prove that (x, y) satisfies constraints (2.2) and (2.4). Since constraint (2.2) is the same as (2.8), we only need to show that (x, y) satisfies (2.4).

For any (ℓ, S) inequality in (2.4), we have

$$\begin{aligned}
& \sum_{i \in S} x_i + \sum_{i \in \mathcal{P}(\ell) \setminus S} d_{i\ell} y_i \\
\geq \; & \sum_{i \in S} \sum_{j \in \mathcal{P}(i,\ell)} q_{ij} + \sum_{i \in \mathcal{P}(\ell) \setminus S} \sum_{j \in \mathcal{P}(i,\ell)} q_{ij} \\
= \; & \sum_{i \in \mathcal{P}(\ell)} \sum_{j \in \mathcal{P}(i,\ell)} q_{ij} \\
= \; & \sum_{j \in \mathcal{P}(\ell)} \sum_{i \in \mathcal{P}(j)} q_{ij} \\
= \; & \sum_{j \in \mathcal{P}(\ell)} d_j,
\end{aligned}$$

where the first inequality follows from (2.5) and (2.7) and the last equality follows from (2.6). Thus, (x, y) satisfies all of the (ℓ, S) inequalities in (2.4). □

To show the reverse direction, for any $(x, y) \in X_{\text{SLSL}}$, we first generate q_{ij} for each $i \in \mathcal{V}$ and $j \in \mathcal{V}(i)$ from x_i using Algorithm 1 given below. Then, we prove that constraints (2.5)–(2.9) are satisfied. For each node j in the scenario tree, we define $a(j)$ to be the unique parent of node j and $\mathcal{C}(j) = \{i \in \mathcal{V} : a(i) = j\}$. Sometimes, we use $c(j)$ as an index for an arbitrary element of $\mathcal{C}(j)$. Now we prove that Algorithm 1 provides a solution q_{ij} such

Algorithm 1 Converting x_i to q_{ij}

.

initialize $t(0) = 1$ and $T = \max\{t(j) : j \in \mathcal{V}\}$.
for $t = 1$ to T, **do**
  let $\Omega(t) = \{j \in \mathcal{V} : t(j) = t\}$.
  while $\Omega(t) \neq \emptyset$ **do**
    select one node $j \in \Omega(t)$ and generate q_{ij} from x_i for each $i \in \mathcal{P}(j)$ as follows.
    let

$$\upsilon_i^j = x_i - \textstyle\sum_{k \in \mathcal{P}(a(j))} q_{ik} \tag{2.11}$$

    for each $i \in \mathcal{P}(j)$ and $\kappa^j = \sum_{i \in \mathcal{P}(j)} \upsilon_i^j$.
    let

$$\sigma_{ij} = d_j \upsilon_i^j / \kappa^j \tag{2.12}$$

    and

$$e(j) = \{i \in \mathcal{P}(j) : \sigma_{ij}/d_j > y_i\}.$$

    if $e(j) = \emptyset$ **then**
      let $q_{ij} = \sigma_{ij}$ for each $i \in \mathcal{P}(j)$.
    else $\{e(j) \neq \emptyset\}$
      while $\sigma_{ij}/d_j > y_i$ for some $i \in \mathcal{P}(j)$ **do**
        define

$$\delta^j = \textstyle\sum_{i \in e(j)} (\sigma_{ij} - d_j y_i) \tag{2.13}$$

        and set $\sigma_{ij} = q_{ij} = d_j y_i$ for each $i \in e(j)$.
        For each $i \in \mathcal{P}(j) \setminus e(j)$, let

$$\tau_i^j = \upsilon_i^j - \sigma_{ij} \tag{2.14}$$

        and

$$\sigma_{ij} = \sigma_{ij} + \delta^j \tau_i^j / \textstyle\sum_{i \in \mathcal{P}(j) \setminus e(j)} \tau_i^j. \tag{2.15}$$

        update $e(j) = e(j) \cup \{i \in \mathcal{P}(j) \setminus e(j) : \sigma_{ij}/d_j > y_i\}$.
      end while
      let $q_{ij} = \sigma_{ij}$ for each $i \in \mathcal{P}(j) \setminus e(j)$.
    end if
    update $\Omega(t) = \Omega(t) \setminus \{j\}$.
  end while
end for

that (x, y, q) is also a feasible solution of RSLS.

Proposition 2.2 *At each iteration of Algorithm 1, $\sum_{i\in\mathcal{P}(j)}\sigma_{ij}=d_j$ for each $j\in\mathcal{V}$.*

Proof: Observe that $\sum_{i\in\mathcal{P}(j)}\sigma_{ij}=d_j$ holds initially in (2.12). During the updating process in (2.15), the total increment of σ_{ij} for each $i\in\mathcal{P}(j)\setminus e(j)$ is

$$\sum_{i\in\mathcal{P}(j)\setminus e(j)}\delta^j\tau_i^j/\sum_{i\in\mathcal{P}(j)\setminus e(j)}\tau_i^j=\delta^j,$$

which is the same as the total decrement of σ_{ij} for each $i\in e(j)$ by the definition of δ^j. Therefore, $\sum_{i\in\mathcal{P}(j)}\sigma_{ij}=d_j$ is maintained at each iteration in (2.15) and the conclusion holds. □

Proposition 2.3 *In Algorithm 1, $q_{ij}/d_j\geq q_{ic(j)}/d_{c(j)}$ for each $i\in\mathcal{P}(j)$ and $j\in\mathcal{V}$.*

Proof: The conclusion is true if $q_{ij}/d_j=y_i$ since we always keep $q_{ij}/d_j\leq y_i$ in Algorithm 1. Now, assume there exists a node $i\in S(j)$ where $S(j)=\{i\in\mathcal{P}(j):q_{ij}/d_j<y_i\}$ such that $q_{ij}/d_j<q_{ic(j)}/d_{c(j)}$. Without loss of generality, assume i to be the first node in $S(j)$ that reaches $q_{ij}/d_j<\sigma_{ic(j)}/d_{c(j)}$ at some iteration in (2.12) or (2.15). Then

(i) For each $k\in S(j)$, considering the construction process for node j, we have $\sigma_{kj}/\sigma_{ij}=\upsilon_k^j/\upsilon_i^j$ at the initial assignment in (2.12) and the increment ratio at each step (2.15) is

$$\rho_{kj}/\rho_{ij}=\tau_k^j/\tau_i^j=(\upsilon_k^j-\sigma_{kj})/(\upsilon_i^j-\sigma_{ij})=\upsilon_k^j/\upsilon_i^j,$$

where $\rho_{kj}=\delta^j\tau_k^j/\sum_{i\in\mathcal{P}(j)\setminus e(j)}\tau_i^j$ and $\rho_{ij}=\delta^j\tau_i^j/\sum_{i\in\mathcal{P}(j)\setminus e(j)}\tau_i^j$. Then we have

$$q_{kj}/q_{ij}=\upsilon_k^j/\upsilon_i^j. \tag{2.16}$$

Note that $k\notin e(c(j))$ for each $k\in S(j)$ before i reaches $q_{ij}/d_j<\sigma_{ic(j)}/d_{c(j)}$ since i is the first node in $S(j)$ that reaches $q_{ij}/d_j<\sigma_{ic(j)}/d_{c(j)}$. Then for the initial setting of $\sigma_{ic(j)}$ for node $c(j)$, we have

$$\sigma_{kc(j)}/\sigma_{ic(j)}=\upsilon_k^{c(j)}/\upsilon_i^{c(j)}=(\upsilon_k^j-q_{kj})/(\upsilon_i^j-q_{ij})=q_{kj}/q_{ij}, \tag{2.17}$$

where the first equality follows from (2.12) and the third inequality follows from (2.16). At each step in (2.15) for node $c(j)$ before $\sigma_{ic(j)}$ reaches $q_{ij}/d_j<\sigma_{ic(j)}/d_{c(j)}$, we have

$$\rho_{kc(j)}/\rho_{ic(j)}=\tau_k^{c(j)}/\tau_i^{c(j)}=(\upsilon_k^{c(j)}-\sigma_{kc(j)})/(\upsilon_i^{c(j)}-\sigma_{ic(j)})=\sigma_{kc(j)}/\sigma_{ic(j)}=q_{kj}/q_{ij},$$

where the third and the fourth equalities follow from (2.17).

Therefore, before $\sigma_{ic(j)}$ hits the value $d_{c(j)}y_i$, we always have $\sigma_{kc(j)}$ nondecreasing and

$$\sigma_{kc(j)}/\sigma_{ic(j)} = q_{kj}/q_{ij}. \tag{2.18}$$

Then, if $q_{ij}/d_j < \sigma_{ic(j)}/d_{c(j)}$, we have

$$q_{kj}/d_j = q_{ij}\sigma_{kc(j)}/(d_j\sigma_{ic(j)}) < \sigma_{ic(j)}\sigma_{kc(j)}/(d_{c(j)}\sigma_{ic(j)}) = \sigma_{kc(j)}/d_{c(j)},$$

where the first equality follows from (2.18). That is,

$$q_{kj}/d_j < \sigma_{kc(j)}/d_{c(j)} \tag{2.19}$$

for each $k \in S(j)$ at the iteration that $\sigma_{ic(j)}$ reaches $q_{ij}/d_j < \sigma_{ic(j)}/d_{c(j)}$.

(ii) For $k \in \mathcal{P}(j) \setminus S(j)$, we have $q_{kj}/d_j = y_k$. Then,

$$k \in e(c(j)) \text{ or } \sigma_{kc(j)}/d_{c(j)} \geq y_k \tag{2.20}$$

at the same iteration $\sigma_{ic(j)}$ reaches $q_{ij}/d_j < \sigma_{ic(j)}/d_{c(j)}$. Otherwise, if $\sigma_{kc(j)}/d_{c(j)} < y_k$ at this iteration, we have

$$\sigma_{kc(j)}/\sigma_{ic(j)} = \upsilon_k^{c(j)}/\upsilon_i^{c(j)} = (\upsilon_k^j - q_{kj})/(\upsilon_i^j - q_{ij}) \geq q_{kj}/q_{ij} \tag{2.21}$$

since $q_{kj}/q_{ij} \leq \upsilon_k^j/\upsilon_i^j$, which follows from the fact that $k \in e(j)$ and $i \in S(j)$. Then,

$$y_k = q_{kj}/d_j \leq \sigma_{kc(j)}q_{ij}/\sigma_{ic(j)}d_j < \sigma_{kc(j)}/d_{c(j)},$$

where the first inequality follows from (2.21) and the second inequality follows from the assumption that $q_{ij}/d_j < \sigma_{ic(j)}/d_{c(j)}$, which is a contradiction.

Combining (2.19) and (2.20), we have

$$\sum_{k\in\mathcal{P}(j)} \sigma_{kc(j)}/d_{c(j)} > \sum_{k\in\mathcal{P}(j)} q_{kj}/d_j,$$

which contradicts the fact that $\sum_{k\in\mathcal{P}(j)} q_{kj}/d_j = 1$ and $\sum_{k\in\mathcal{P}(j)} \sigma_{kc(j)}/d_{c(j)} = 1 - \sigma_{c(j)c(j)}/d_{c(j)} \leq 1$. Therefore, the claim holds. $\square$

Proposition 2.4 $X_{\mathrm{SLSL}} \subseteq \mathrm{proj}_{\{\mathrm{x,y}\}}(\mathrm{X_{RSLS}})$.

Proof: For any $(x, y) \in X_{\mathrm{SLSL}}$, we only need to show that q_{ij} for each $i \in \mathcal{V}$ and $j \in \mathcal{V}(i)$ obtained from x_i by Algorithm 1 satisfies constraints (2.5)–(2.9). Constraints (2.8) are the same as (2.2) and each q_{ij} is non-negative in the algorithm. Thus, it remains to prove that constraints (2.5)–(2.7) are satisfied. Proposition 2.2 implies that constraints (2.6) are satisfied. Constraints (2.7) are satisfied since $q_{ij}/d_j \leq y_i$ always holds in Algorithm 1. Now we consider constraints (2.5). We have

(a) $\sigma_{ij} \leq \upsilon_i^j$ in (2.12). That is, $d_j \leq \kappa^j$. Otherwise if $\kappa^j < d_j$, then $\kappa^j = \sum_{i\in\mathcal{P}(j)} \upsilon_i^j = \sum_{i\in\mathcal{P}(j)} x_i - \sum_{i\in\mathcal{P}(j)} \sum_{k\in\mathcal{P}(a(j))} q_{ik} = \sum_{i\in\mathcal{P}(j)} x_i - \sum_{i\in\mathcal{P}(a(j))} d_i < d_j$, which contradicts $\sum_{i\in\mathcal{P}(j)} x_i \geq \sum_{i\in\mathcal{P}(j)} d_i$.

(b) $\sigma_{ij} \leq \upsilon_i^j$ in (2.15). That is, $\delta^j \leq \sum_{i\in\mathcal{P}(j)\backslash e(j)} \tau_i^j$. Otherwise suppose that $\delta^j > \sum_{i\in\mathcal{P}(j)\backslash e(j)} \tau_i^j$. Note that for each $i \in e(j)$, we have $q_{ik} = y_i d_k$ for each $k \in \mathcal{P}(i, a(j))$ by Proposition 2.3. Then,

$$\begin{aligned}
d_{0j} &= \sum_{i\in\mathcal{P}(j)} \sum_{k\in\mathcal{P}(j)} \sigma_{ik} \\
&= \sum_{i\in e(j)} d_{ij} y_i + \sum_{i\in e(j)} (\sigma_{ij} - d_j y_i) + \sum_{i\in\mathcal{P}(j)\backslash e(j)} \sum_{k\in\mathcal{P}(j)} \sigma_{ik} \\
&= \sum_{i\in e(j)} d_{ij} y_i + \delta^j + \sum_{i\in\mathcal{P}(j)\backslash e(j)} \sum_{k\in\mathcal{P}(j)} \sigma_{ik} \\
&= \sum_{i\in e(j)} d_{ij} y_i + \delta^j + \sum_{i\in\mathcal{P}(j)\backslash e(j)} (x_i - \upsilon_i^j + \sigma_{ij}) \\
&= \sum_{i\in e(j)} d_{ij} y_i + \delta^j + \sum_{i\in\mathcal{P}(j)\backslash e(j)} (x_i - \tau_i^j)
\end{aligned}$$

where the third equality follows from (2.13), the fourth equality follows from (2.11) and the fifth equality follows from (2.14). Then,

$$\begin{aligned}
d_{0j} = &\sum_{i\in e(j)} d_{ij} y_i + \sum_{i\in\mathcal{P}(j)\backslash e(j)} x_i + \delta^j - \sum_{i\in\mathcal{P}(j)\backslash e(j)} \tau_i^j \\
> &\sum_{i\in e(j)} d_{ij} y_i + \sum_{i\in\mathcal{P}(j)\backslash e(j)} x_i,
\end{aligned}$$

which contradicts the validity of the corresponding (ℓ, S) inequality.

Based on (a) and (b), constraints (2.5) are satisfied. $\square$

Combining Propositions 2.1 and 2.4, we obtain

Theorem 2.2 $X_{\text{SLSL}} = \text{proj}_{\{\text{x,y}\}}(\text{X}_{\text{RSLS}})$.

Theorem 2.2 can be easily extended to generalizations of the stochastic uncapacitated lot-sizing problem. For example, if we include capacities of the form $x_i \leq c_i y_i$, or start-up costs of the form $y_i \leq y_{a(i)} + z_i$ where $z_i = 1$ iff there is production at node i but not at the node that precedes i, the results of Theorem 2.2 regarding the equality of the LP bounds are still valid.

CHAPTER 3

TWO-PERIOD STOCHASTIC UNCAPACITATED LOT-SIZING PROBLEM

3.1 Introduction

In Chapter 2, we described a stochastic uncapacitated lot-sizing problem formulation and showed that the (ℓ, S) inequalities are valid for it. We then provided another formulation and proved the equivalence between the two formulations. In Section 3.2 of this chapter, we provide an example to show that neither of these formulations can guarantee optimal integral solutions even for two-period stochastic uncapacitated lot-sizing problems. Then, in Section 3.3, we analyze the two-period stochastic uncapacitated lot-sizing problem and provide a complete description of the convex hull of feasible integer solutions. We also show that the corresponding separation algorithm runs in polynomial time. Finally, we conclude with a few remarks in Section 3.4. The results of this chapter also appear in [45].

3.2 An Example

As presented in Section 1.4, adding all the (ℓ, S) inequalities is sufficient to describe the convex hull of integer solutions in the deterministic case. However, this is not true for the stochastic case. The following example shows that there are fractional optimal solutions after we add all of the (ℓ, S) inequalities even for a two-period problem.

Example: Consider an instance with four nodes. Node 0 is the root node. Nodes 1, 2 and 3 are three successors of node 0. The problem parameters are:

1. $p_0 = 1, p_1 = 1/3, p_2 = 1/3$ and $p_3 = 1/3$.
2. $d_0 = 20$, $d_1 = 30$, $d_2 = 40$ and $d_3 = 50$.
3. $h_0 = 10$ and $h_1 = h_2 = h_3 = 0$.

4. $\alpha_0 = 190$, $\alpha_1 = 1000$, $\alpha_2 = 10$ and $\alpha_3 = 35$.

5. $\beta_0 = 10$, $\beta_1 = 1000$, $\beta_2 = 100$ and $\beta_3 = 100$.

The optimal objective value of the LP relaxation of SLS with all of the (ℓ, S) inequalities added is 10831.6 and the corresponding optimal solution is $x_0 = 50, x_1 = 0, x_2 = 10$ and $x_3 = 20$, $y_0 = 1, y_1 = 0, y_2 = 0.25$ and $y_3 = 0.4$. However, the optimal objective value of SLS is 10876.6 with the corresponding optimal solution $x_0 = 50, x_1 = 0, x_2 = 10$ and $x_3 = 20$, $y_0 = 1, y_1 = 0, y_2 = 1$ and $y_3 = 1$. $\square$

3.3 Convex Hull Results

For the two-period SLS, every node except the root node 0 is in $\mathcal{C}(0)$. Let $s_0 = x_0 - d_0$ so that we can formulate the two-period SLS as

$$\text{(SLS2)}: \min \quad h_0 s_0 + \textstyle\sum_{i \in \mathcal{C}(0)} p_i(\alpha_i x_i + \beta_i y_i)$$

$$s_0 + x_i \geq d_i \qquad i \in \mathcal{C}(0) \tag{3.1}$$

$$x_i \leq d_i y_i \qquad i \in \mathcal{C}(0) \tag{3.2}$$

$$s_0 \geq 0 \text{ and } x_i \geq 0 \qquad i \in \mathcal{C}(0) \tag{3.3}$$

$$y_i \in \{0, 1\} \qquad i \in \mathcal{C}(0). \tag{3.4}$$

Without loss of generality, we assume that $d_1 < \ldots < d_{|\mathcal{C}(0)|}$.

In the remainder of this section, we search for a family of inequalities that can describe $\text{conv}(X_{\text{SLS2}})$. We consider the extreme points of $\text{conv}(X_{\text{SLS2}})$. In each of these points, the value of s_0 is determined by some demand $d_i,\ i \in \mathcal{C}(0)$. This yields the result:

Proposition 3.1 *Given an extreme point (x, y, s_0) of $\text{conv}(X_{\text{SLS2}})$, let $i' = \text{argmax}_i\{d_i : s_0 \geq d_i\}$. Then*

1. *the equations $s_0 = d_{i'}$ and $x_{i'} = 0$ hold;*

2. *for each $i \in \mathcal{C}(0) : i > i'$, either (a) $x_i = d_i - d_{i'}$ and $y_i = 1$ or (b) $x_i = d_i$ and $y_i = 1$;*

3. *for each* $i \in \mathcal{C}(0) : i \leq i'$, *(a)* $x_i = 0$ *and* $y_i = 0$, *or (b)* $x_i = 0$ *and* $y_i = 1$, *or (c)* $x_i = d_i$ *and* $y_i = 1$.

Note that for $i \in \mathcal{C}(0) : i \leq i'$, y_i can take the value of either 0 or 1 in an extreme point. Also, for each $i \in \mathcal{C}(0)$ for which $y_i = 1$, x_i can take either of the values $\max\{d_i - d_{i'}, 0\}$ or d_i. Because of this, the number of extreme points is exponential.

However, the fact that there are only $|\mathcal{C}(0)|$ possible optimal values of s_0 means that we can optimize over X_{SLS2} in polynomial time. For each extreme point for which $s_0 = d_{i'}$, the variables $y_i, i \in \mathcal{C}(0)$ such that $i > i'$, are fixed to 1 by this choice of s_0. For each x_i such that $i > i'$, if $\alpha_i \geq 0$, then there is an optimal solution in which $x_i = d_i - d_{i'}$; otherwise, $x_i = d_i$. The optimal solution values for $i \in \mathcal{C}(0)$ such that $i \leq i'$ can be similarly identified. This leads to an extended formulation.

We introduce variables $\lambda_i, i \in \mathcal{C}(0)$, one for each possible nonzero extreme point value of s_0. Also, we introduce the dummy variable λ_o to represent the extreme point in which $s_0 = 0$, and set $d_o = 0$ where we use o to differentiate this subscript from the root node 0. Let (SLS2λ) be the formulation defined by the constraints

$$\textstyle\sum_{i \in \mathcal{C}(0) \cup o} \lambda_i = 1 \tag{3.5}$$

$$s_0 \geq \textstyle\sum_{i \in \mathcal{C}(0)} d_i \lambda_i \tag{3.6}$$

$$y_i \geq \textstyle\sum_{j \in \mathcal{C}(0):j<i} \lambda_j + \lambda_o, \qquad i \in \mathcal{C}(0) \tag{3.7}$$

$$x_i \geq \textstyle\sum_{j \in \mathcal{C}(0):j<i} (d_i - d_j)\lambda_j + d_i \lambda_o, \quad i \in \mathcal{C}(0) \tag{3.8}$$

$$x_i \leq d_i y_i, \qquad i \in \mathcal{C}(0) \tag{3.9}$$

$$\lambda_i \geq 0, \qquad i \in \mathcal{C}(0) \cup o \tag{3.10}$$

$$y_i \leq 1, \qquad i \in \mathcal{C}(0). \tag{3.11}$$

Let $X_{\text{SLS2}\lambda}$ be the set of feasible points to this formulation. Also, let $X_{\text{SLS2}\lambda-\text{x}}$ be the set of feasible points to (3.5)–(3.7), (3.10)–(3.11) in the (y, s_0, λ) space.

Proposition 3.2 *Every extreme point of* $X_{\text{SLS2}\lambda-\text{x}}$ *is integral in* y *and* λ. *Moreover, in* *every extreme point*

$$d_i\lambda_o + \sum_{j\in\mathcal{C}(0):j<i} (d_i - d_j)\lambda_j \le d_i y_i, i \in \mathcal{C}(0).$$

Proof: In any extreme point of $X_{\text{SLS2}\lambda}$, we first observe that (3.6) must hold at equality, and so s_0 is determined by λ. We also observe that y_i can be either 1 or $\sum_{j\in\mathcal{C}(0):j<i}\lambda_j + \lambda_o$ since $\sum_{j\in\mathcal{C}(0):j<i}\lambda_j + \lambda_o \le \sum_{i\in\mathcal{C}(0)\cup o}\lambda_i = 1$, where the equality follows from (3.5). Since the coefficient matrix corresponding to λ in (3.5) and (3.7) has the consecutive ones property, we have that every extreme point of $X_{\text{SLS2}\lambda-\text{x}}$ is integral in y and λ. The condition $d_i\lambda_o + \sum_{j\in\mathcal{C}(0):j<i}(d_i - d_j)\lambda_j \le d_i y_i, i \in \mathcal{C}(0)$ follows from the condition $d_i\lambda_o + \sum_{j\in\mathcal{C}(0):j<i}(d_i - d_j)\lambda_j \le d_i\lambda_o + \sum_{j\in\mathcal{C}(0):j<i} d_i\lambda_j \le d_i y_i, i \in \mathcal{C}(0)$, where the second inequality follows from (3.7). □

The previous proposition shows that introducing the x variables and the constraints (3.8)–(3.9), which gives $X_{\text{SLS2}\lambda}$, does not create any fractional extreme points, since these constraints simply require that a single variable falls between two bounds.

Proposition 3.3 *If (x, y, s, λ) is an extreme point of $X_{\text{SLS2}\lambda}$, then $(x, y, s) \in X_{\text{SLS2}}$.*

Proof: From the proof of Proposition 3.2, for each extreme point of $X_{\text{SLS2}\lambda}$, we have that (3.6) holds at equality and so s_0 is determined by λ, and for each $i \in \mathcal{C}(0)$, either (3.7) holds at equality, or $y_i = 1$ (perhaps both). Thus, each y_i is either determined by λ or set equal to 1. Similarly, x_i is determined either by (3.8) or (3.9) for each $i \in \mathcal{C}(0)$.

Thus, all the variables except for λ are either set to an upper bound or depend on λ in each extreme point. The only constraints that actually impose restrictions on λ are (3.5) and (3.10).

Based on Proposition 3.2, the extreme points are integral in λ and y. Assuming there exists an $i' \in \mathcal{C}(0)$ such that $\lambda_{i'} = 1$ and $\lambda_i = 0$ for all $i \in \mathcal{C}(0) \cup o \setminus \{i'\}$, we have $s_0 = d_{i'}$, $y_i = 1$ and $d_i \ge x_i \ge d_i - d_{i'}$ for each $i > i'$ from (3.6)–(3.9). Then, the result follows by simply checking that (3.1)–(3.4) hold. □

The next proposition follows immediately from Proposition 3.1.

Proposition 3.4 *If (x, y, s) is an extreme point of* $\text{conv}(X_{\text{SLS2}})$*, then (x, y, s, λ) is a feasible solution of (3.5)–(3.11), where λ is defined by choosing i' so that $s_0 = d_{i'}$ and setting $\lambda_{i'} = 1$, $\lambda_i = 0$, $i \in \mathcal{C}(0) \cup o \setminus i'$.*

The previous two propositions show

Proposition 3.5 $\text{Proj}_{\{x,y,s\}}(X_{\text{SLS2}\lambda}) = X_{\text{SLS2}}$.

By using this proposition and Farkas' Lemma, we can define additional inequalities, besides (3.1)–(3.3) and (3.11), that are needed to define $\text{conv}(X_{\text{SLS2}})$ in the (x, y, s) space. To do this, consider any point $(\bar{x}, \bar{y}, \bar{s}_0)$ satisfying (3.1)–(3.3) and (3.11). Then by Proposition 3.5 this point is in $\text{conv}(X_{\text{SLS2}})$ if and only if there exists a $\bar{\lambda}$ such that $(\bar{x}, \bar{y}, \bar{s}_0, \bar{\lambda})$ satisfies (3.5)–(3.11).

By associating dual variables μ with constraint (3.5), ν with (3.6), ϕ with (3.7), and ρ with (3.8), Farkas' Lemma implies that $(\bar{x}, \bar{y}, \bar{s}_0) \in \text{conv}(X_{\text{SLS2}})$ if and only if the dual cone of (3.5)–(3.11), given by (3.13)–(3.15) does not have an extreme ray that makes the objective function (3.12) unbounded:

$$\text{(DSLS)} : \max \quad \mu - \nu \bar{s}_0 - \sum_{i \in \mathcal{C}(0)} \phi_i \bar{y}_i - \sum_{i \in \mathcal{C}(0)} \rho_i \bar{x}_i \tag{3.12}$$

$$\mu - d_i \nu - \sum_{j \in \mathcal{C}(0): d_j > d_i} \phi_j - \sum_{j \in \mathcal{C}(0): d_j > d_i} (d_j - d_i) \rho_j \leq 0, i \in \mathcal{C}(0), \tag{3.13}$$

$$\mu - \sum_{i \in \mathcal{C}(0)} \phi_i - \sum_{i \in \mathcal{C}(0)} d_i \rho_i \leq 0, \tag{3.14}$$

$$\nu \geq 0; \;\; \phi_i \geq 0, \rho_i \geq 0, i \in \mathcal{C}(0). \tag{3.15}$$

Here, constraints (3.13) correspond to $\lambda_i, i \in \mathcal{C}(0)$, and constraint (3.14) corresponds to λ_o. Restating, we obtain the following Proposition

Proposition 3.6 *If $(\bar{x}, \bar{y}, \bar{s}_0)$ satisfies (3.1)–(3.3) and (3.11), then $(\bar{x}, \bar{y}, \bar{s}_0) \in$* $\text{conv}(X_{\text{SLS2}})$ *if and only if*

$$\nu \bar{s}_0 + \textstyle\sum_{i \in \mathcal{C}(0)} \phi_i \bar{y}_i + \sum_{i \in \mathcal{C}(0)} \rho_i \bar{x}_i \geq \mu \tag{3.16}$$

for all extreme rays of (3.13)–(3.15).

We now only need to show that the optimal objective value of DSLS is not greater than zero, which implies that the corresponding inequalities will be the only ones needed to be added to describe $\text{conv}(X_{\text{SLS2}})$. Given a point $(\bar{x}, \bar{y}, \bar{s}_0)$ satisfying (3.1)–(3.3) and (3.11), we see that no dual ray in which $\nu = 0$ can ever violate (3.16). This is because the constraint (3.13) for $i_{|\mathcal{C}(0)|}$ implies that $\mu \leq 0$. Therefore, the right-hand side of (3.16) cannot exceed the left.

We normalize dual rays in which $\nu > 0$ so that $\nu = 1$. Given this normalization, we show that (3.12)–(3.15) imply that the ray maximizing (3.12), and hence yielding the most violated inequality of the form (3.16), has the properties shown in Proposition 3.7. In the following proofs, constraint (3.14) can be included in constraint (3.13) since we assume $d_o = 0$.

Proposition 3.7 *Given that $\nu = 1$, there exists a ray that maximizes (3.12) in which $\mu = d_{|\mathcal{C}(0)|}$.*

Proof: Given $\nu = 1$, constraint (3.13) for $i = |\mathcal{C}(0)|$ defines an upper bound of $d_{|\mathcal{C}(0)|}$ for μ. If $\mu < d_{|\mathcal{C}(0)|}$, we can always increase μ and $\phi_{|\mathcal{C}(0)|}$ by $d_{|\mathcal{C}(0)|} - \mu > 0$, which is feasible to DSLS with a non-decreasing objective value since $\bar{y}_{|\mathcal{C}(0)|} \leq 1$. □

Proposition 3.8 *Given that $\nu = 1$, there exists a ray that maximizes (3.12) in which*

$$\textstyle\sum_{i \in \mathcal{C}(0)} \rho_i \leq 1. \tag{3.17}$$

Proof: Proof by induction.

Initial step: First we show that $\rho_{|\mathcal{C}(0)|} \leq 1$. If not, we can decrease $\rho_{|\mathcal{C}(0)|}$ to 1, which is feasible to DSLS with a non-decreasing objective value since $\bar{x}_{|\mathcal{C}(0)|} \geq 0$ and $\mu = d_{|\mathcal{C}(0)|}$ by Proposition 3.7.

Recursive step: Assume $\sum_{i \in \mathcal{C}(0): i \geq j} \rho_i \leq 1$ is satisfied by some optimal solution. We need to show that $\sum_{i \in \mathcal{C}(0): i \geq j-1} \rho_i \leq 1$ is also satisfied by this optimal solution. If $\sum_{i \in \mathcal{C}(0): i \geq j-1} \rho_i > 1$ in the optimal solution, then we have $\rho_{j-1} > 1 - \sum_{i \in \mathcal{C}(0): i \geq j} \rho_i \geq 0$ based on the induction hypothesis. Decreasing ρ_{j-1} to be $1 - \sum_{i \in \mathcal{C}(0): i \geq j} \rho_i$ in the optimal solution generates a non-decreasing objective value. We only need to check if the constraints (3.13) for

$i \in \mathcal{C}(0), i \leq j-1$ are satisfied after updating ρ_{j-1}. Since $\phi_{j-1} \geq 0$, the left-hand side of the constraints (3.13) for j is less than or equal to zero, and $\sum_{i \in \mathcal{C}(0): i \geq j-1} \rho_i = 1$ after updating ρ_{j-1}, the left-hand side of the constraints (3.13) for $j-1$ is equal to

$$\begin{aligned} &\text{(the left-hand side of the constraints (3.13) for } j) \\ &+(d_j - d_{j-1}) - \phi_{j-1} - (d_j - d_{j-1}) \sum_{i \in \mathcal{C}(0): i \geq j-1} \rho_i \end{aligned} \tag{3.18}$$

and is less than or equal to zero. Similar arguments prove that constraints (3.13) for $i < j-1$ are satisfied by the updated solution. $\square$

Let $i^* = \text{argmax}\{i \in \mathcal{C}(0) : \bar{y}_i = 0\}$ or $i^* = o$ if $\bar{y}_i > 0$ for all $i \in \mathcal{C}(0)$. In the following, we first consider the case of $i^* \neq o$. By the definition of i^*, we have $\bar{x}_{i^*} = 0$. By Proposition 3.8, there exists an optimal solution such that

$$\rho_{i^*} = 1 - \textstyle\sum_{i \in \mathcal{C}(0): i > i^*} \rho_i. \tag{3.19}$$

By (3.18), constraints (3.13) for $i \leq i^*$ are satisfied if constraints (3.13) for $i > i^*$ are satisfied. This implies that there exists an optimal solution such that $\phi_j = \rho_j = 0$ for each $j < i^*$. Now, we only need to find ϕ_j, ρ_j for each $j > i^*$ to satisfy constraints (3.13) for $i > i^*$. If $i^* = |\mathcal{C}(0)|$, then $\tau_{|\mathcal{C}(0)|} = 0$, $\rho_{|\mathcal{C}(0)|} = 1$, and we can set $\rho_i = 0$ for each $i \in \mathcal{C}(0) \setminus |\mathcal{C}(0)|$, $\phi_i = 0$ for each $i \in \mathcal{C}(0)$ and the corresponding optimal objective value is $u - \bar{s}_0 - \bar{x}_{|\mathcal{C}(0)|} \leq 0$, which follows from constraints (3.1). No more inequalities need to be added. Now, assume $o \neq i^* < |\mathcal{C}(0)|$ and only consider optimal solution values ϕ_j and ρ_j for all $j \geq i^*$. Define

$$\mathcal{Q} = \{i \in \mathcal{C}(0) : i \geq i^* \text{ and } \phi_i > 0\} \cup i^* = \{i_1, i_2, \ldots, i_Q\}. \tag{3.20}$$

Note here $i^* = i_1$. Let $\rho_i = \phi_i = 0$ for all $i \notin \mathcal{Q}$. Now we claim that

Proposition 3.9 *Given that $\nu = 1$, there exists a ray that maximizes (3.12) for which $\bar{y}_{i_1} \leq \ldots \leq \bar{y}_{i_Q}$.*

Proof: If $\bar{y}_{i_1} > \bar{y}_{i_2}$, then there exists an $\epsilon > 0$ such that ϕ_{i_1} can be decreased and ϕ_{i_2} can be increased by ϵ. The new solution is feasible with a better objective value. $\square$

Proposition 3.10 *Given that $\nu = 1$, there exists a ray that maximizes (3.12) and for which, for a given $\mathcal{Q}$ defined by (3.20), we have*

$$\mu = d_{|\mathcal{C}(0)|}, \rho_{i^*} = 1, \phi_{i^*} = 0$$

$$\phi_{i_j} = d_{i_j} - d_{i_{j-1}}, \qquad j = 2, \ldots, Q$$

$$\phi_i = 0, \qquad i \in \mathcal{C}(0) \setminus \mathcal{Q}.$$

$$\rho_i = 0, \qquad i \in \mathcal{C}(0) \setminus \{i^*\}.$$

Proof: In the proof of Propositions 3.7 and 3.8, we have shown that $\phi_i = \rho_i = 0$ for each $i \in \mathcal{C}(0) \setminus \mathcal{Q}$ and $\mu = d_{|\mathcal{C}(0)|}$. Now, we show that the optimal value ϕ_i is a function of ρ_i and the optimal objective value is also a function of ρ_i for each $i \in \mathcal{C}(0)$. We consider two cases:

Case (1): $\phi_i > 0$ for all $i > i^*$. Based on Proposition 3.9, we have $\bar{y}_{i^*+1} \leq \ldots \leq \bar{y}_{|\mathcal{C}(0)|}$. Thus

$$\phi_i = (1 - \textstyle\sum_{j \geq i} \rho_j)(d_i - d_{i-1}) \tag{3.21}$$

for each $i > i^*$ starting from $\phi_{|\mathcal{C}(0)|}$. Then the optimal objective value is

$$Z_{\text{DSLS}} = d_{|\mathcal{C}(0)|} - \bar{s}_0 - \rho_{i^*}\bar{x}_{i^*} - \textstyle\sum_{i>i^*}(d_i - d_{i-1})\bar{y}_i - \sum_{i>i^*}\rho_i(\bar{x}_i - \sum_{i \geq j > i^*}(d_j - d_{j-1})\bar{y}_j).$$

We will see that $\bar{x}_i - \sum_{i \geq j > i^*}(d_j - d_{j-1})\bar{y}_j < 0$ for all $i > i^*$. Otherwise, if there exists an $i' > i^*$ such that $\bar{x}_{i'} - \sum_{i' \geq j > i^*}(d_j - d_{j-1})\bar{y}_j > 0$, we will have $\rho_{i'} = 1$ for the optimal solution and then $\phi_{i^*+1} = 0$ by (3.21), which contradicts the assumption that $\phi_i > 0$ for each $i > i^*$. Then, we have $\rho_i = 0$ for each $i > i^*$, $\rho_{i^*} = 1$ and

$$\phi_i = d_i - d_{i-1}, \ \forall i > i^*.$$

The corresponding optimal objective value is $d_{|\mathcal{C}(0)|} - \bar{s}_0 - \bar{x}_{i^*} - \sum_{i>i^*}(d_i - d_{i-1})\bar{y}_i$ and the corresponding inequality

$$d_{|\mathcal{C}(0)|} \leq s_0 + x_{i^*} + \textstyle\sum_{i>i^*}(d_i - d_{i-1})y_i$$

is needed to describe the convex hull of SLS2.

Case (2): There exists a $k > i^*$ such that $\phi_k = 0$. If $\phi_{|\mathcal{C}(0)|} = 0$, then $\rho_{|\mathcal{C}(0)|} = 1$ follows from the constraint (3.13) for $i = |\mathcal{C}(0)| - 1$ and all remaining constraints (3.13) are satisfied.

The corresponding optimal objective value is $d_{|\mathcal{C}(0)|} - \bar{s}_0 - \bar{x}_{|\mathcal{C}(0)|}$, which is less than or equal to zero since the corresponding inequality $s_0 + x_{|\mathcal{C}(0)|} \geq d_{|\mathcal{C}(0)|}$ is valid in the original SLS2.

Since $\phi_j = 0$ for each $i_{k-1} < j < i_k$, if constraint (3.13) for $i = i_{k-1}+1$ is satisfied, then constraints (3.13) for $i_{k-1}+1 < j \leq i_k$ are satisfied by Proposition 3.8 and the recursive step (3.18). Then, following the similar argument in case (1) with the only difference being that ϕ_{i_k} is determined by the constraint (3.13) for $i = i_{k-1}+1$, we have

$$\phi_{i_j} = \textstyle\sum_{i_j \geq k > i_{j-1}} (1 - \sum_{m \geq k} \rho_m)(d_k - d_{k-1}).$$

Following the same argument of case (1), we have $\rho_{i^*} = 1,\ \rho_i = 0\ \forall i > i^*$ and thus $\phi_{i_j} = d_{i_j} - d_{i_{j-1}}$. The corresponding objective value is

$$d_{|\mathcal{C}(0)|} - \bar{s}_0 - \bar{x}_{i^*} - \textstyle\sum_{i_j \in \mathcal{Q}} (d_{i_j} - d_{i_{j-1}})\bar{y}_{i_j},$$

and the corresponding inequality

$$d_{|\mathcal{C}(0)|} \leq s_0 + x_{i^*} + \textstyle\sum_{i_j \in \mathcal{Q}} (d_{i_j} - d_{i_{j-1}}) y_{i_j}$$

is needed to describe the convex hull of X_{SLS2}. □

Remark 3.1 *If $i^* = o$, then by Proposition 3.10, we have $\rho_i = 0$ for all $i \in \mathcal{C}(0)$ and $\phi_{i_j} = d_{i_j} - d_{i_{j-1}}$ for each $i_j \in \mathcal{Q}$. Then, the corresponding objective value is $d_{|\mathcal{C}(0)|} - \bar{s}_0 - \sum_{i_j \in \mathcal{Q}} (d_{i_j} - d_{i_{j-1}})\bar{y}_{i_j}$, and the corresponding inequality is $d_{|\mathcal{C}(0)|} \leq s_0 + \sum_{i_j \in \mathcal{Q}} (d_{i_j} - d_{i_{j-1}}) y_{i_j}$. This inequality is dominated by $d_{|\mathcal{C}(0)|} \leq s_0 + x_{i_1} + \sum_{i_j \in \mathcal{Q} \setminus \{i_1\}} (d_{i_j} - d_{i_{j-1}}) y_{i_j}$.*

Putting these results together yields

Theorem 3.1 *The inequalities*

$$s_0 + x_{i_1} + \sum_{j=2}^{Q} (d_{i_j} - d_{i_{j-1}}) y_{i_j} \geq d_{|\mathcal{C}(0)|}, \tag{3.22}$$

where $\mathcal{Q} = \{i_1, i_2, \ldots, i_Q\} \subseteq \mathcal{C}(0)$ and sorted $d_{i_1} \leq d_{i_2} \leq \ldots \leq d_{i_Q}$ together with (3.1)–(3.3) and (3.11) are the only inequalities needed to define the convex hull of X_{SLS2}.

Proof: The proof follows directly from Proposition 3.10. □

Proposition 3.11 *Given that $\nu = 1$, there exists a ray that maximizes (3.12) in which $\bar{y}_i \geq \bar{y}_{i_j}$ for all i such that $d_{i_{j-1}} < d_i < d_{i_j}$, for $j = 2, \ldots, |\mathcal{C}(0)|$.*

Proof: If not, by Proposition 3.10, we can increase ϕ_i and decrease ϕ_{i_q} by a small positive value $\epsilon > 0$, which is feasible with a better objective value. □

Proposition 3.11 shows us how to separate the inequalities (3.22) defined in Theorem 3.1. Separation can be done by first putting $|\mathcal{C}(0)|$ into $\mathcal{Q}$ and setting $\overline{y}_{min} = \overline{y}_{|\mathcal{C}(0)|}$. Then, we proceed backward through the elements of $\mathcal{C}(0)$: if $\overline{y}_i < \overline{y}_{min}$ for any i, then put i into $\mathcal{Q}$ and set $\overline{y}_{min} = \overline{y}_i$. This procedure will identify a set $\mathcal{Q}$ that defines a violated inequality of the form defined in Theorem 3.1 if one exists.

Proposition 3.12 *The inequalities (3.22) for SLS2 can be separated in $\mathcal{O}(|\mathcal{C}(0)|\log(|\mathcal{C}(0)|))$ time.*

Note that if we do not need to take into account the time needed to sort $i \in \mathcal{C}(0)$ by the demands, then the separation algorithm runs in linear time.

Example (continued): We augment the LP in the example with the inequality (3.22)

$$x_0 + x_1 + 10y_2 + 10y_3 \geq 70.$$

Then, the optimal integral solution is obtained.

3.4 Concluding Remarks

In this chapter, we began our polyhedral study for stochastic lot-sizing problems. We developed a family of inequalities that are sufficient to describe the convex hull of solutions of the two-period stochastic uncapacitated lot-sizing problem. In the next chapter, we study the polyhedron for the multi-period stochastic uncapacitated lot-sizing problem.

CHAPTER 4

MULTI-STAGE STOCHASTIC UNCAPACITATED LOT-SIZING PROBLEM

4.1 Introduction

In this chapter, we study the multi-stage stochastic uncapacitated lot-sizing problem. In Section 4.2, we generalize the (ℓ, S) inequalities to a new class of valid inequalities for the stochastic lot-sizing polytope. Then in Section 4.3, we provide necessary and sufficient conditions that guarantee that the proposed inequalities are facet-defining. In Section 4.4, we develop separation algorithms. Finally, our computational experiments in Section 4.5 demonstrate that the proposed inequalities are extremely useful within a branch-and-cut scheme for stochastic lot-sizing problems. The results of this chapter also appear in [47].

4.2 The $(\mathcal{Q}, S_{\mathcal{Q}})$ Inequalities

In this section, we extend the (ℓ, S) inequalities to a general class called the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequalities. For convenience, we repeat the SLS formulation here.

$$\text{(SLS)}: \quad \min \quad \sum_{i \in \mathcal{V}} (\bar{\alpha}_i x_i + \bar{\beta}_i y_i) \tag{4.1}$$

$$\sum_{j \in \mathcal{P}(i)} x_j \geq d_{0i} \qquad i \in \mathcal{V}, \tag{4.2}$$

$$0 \leq x_i \leq M_i y_i \qquad i \in \mathcal{V}, \tag{4.3}$$

$$y_i \in \{0, 1\} \qquad i \in \mathcal{V}, \tag{4.4}$$

where $\bar{\alpha}_i = p_i \alpha_i + \sum_{j \in \mathcal{V}(i)} p_j h_j$ and $\bar{\beta}_i = p_i \beta_i$.

Consider a subset $\mathcal{Q} \subset \mathcal{V} \setminus \{0\}$ satisfying the following properties:

(A1) If $i, j \in \mathcal{Q}$, then $d_{0i} \neq d_{0j}$.

(A2) If $i, j \in \mathcal{Q}$, then $i \notin \mathcal{P}(j)$ and $j \notin \mathcal{P}(i)$.

(A1) allows us to uniquely index the nodes in the set $\mathcal{Q}$ as $\{1, 2, ..., Q\}$ where $Q = |\mathcal{Q}|$, such that $d_{01} < d_{02} < \cdots < d_{0Q}$. (A2) simply gives us a convenient way of defining the subtrees over which the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequalities are defined. We will comment on (A1) and (A2) at the end of this section.

Define $\mathcal{T}_{\mathcal{Q}} = \{\mathcal{V}_{\mathcal{Q}}, \mathcal{E}_{\mathcal{Q}}\}$ to be the subtree of $\mathcal{T}$ whose leaf nodes are $\mathcal{Q}$, i.e, $\mathcal{V}_{\mathcal{Q}} = \cup_{i \in \mathcal{Q}} \mathcal{P}(i)$. Note that by (A2), *all* nodes in $\mathcal{Q}$ are leaf nodes of $\mathcal{T}_{\mathcal{Q}}$. Given $i \in \mathcal{V}_{\mathcal{Q}}$, we denote by $\mathcal{T}_{\mathcal{Q}}(i) = \{\mathcal{V}_{\mathcal{Q}}(i), \mathcal{E}_{\mathcal{Q}}(i)\}$ the subtree of $\mathcal{T}_{\mathcal{Q}}$ with i as the root node. Note that $\mathcal{V}_{\mathcal{Q}}(i) = \mathcal{V}(i) \cap \mathcal{V}_{\mathcal{Q}}$. We use $\mathcal{Q}(i) \subseteq \mathcal{Q}$ to denote the set of leaf nodes of the subtree $\mathcal{T}_{\mathcal{Q}}(i)$, i.e., $\mathcal{Q}(i) = \mathcal{V}_{\mathcal{Q}}(i) \cap \mathcal{Q}$.

In addition to (A1) and (A2), we need the following property on the set $\mathcal{Q}$ for the validity of the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequalities:

> **(A3)** Given any node $k \in \mathcal{V}_{\mathcal{Q}}$, and nodes $i, j \in \mathcal{Q}$ such that $i < j$ and $i, j \in \mathcal{Q}(k)$, we have that $\{i, i+1, \ldots, j-1, j\} \subseteq \mathcal{Q}(k)$.

Given a subset $\mathcal{Q}$, define the following quantities for all nodes $i \in \mathcal{V}_{\mathcal{Q}}$:

$$\overline{D}_{\mathcal{Q}}(i) = \max\{d_{0j} : j \in \mathcal{Q}(i)\} \tag{4.5}$$

$$\widetilde{D}_{\mathcal{Q}}(i) = \begin{cases} 0, \quad \text{if } \{j : j \in \mathcal{Q} \setminus \mathcal{Q}(i) \text{ such that } d_{0j} \leq \overline{D}_{\mathcal{Q}}(i)\} = \emptyset \\ \max\{d_{0j} : j \in \mathcal{Q} \setminus \mathcal{Q}(i) \text{ such that } d_{0j} \leq \overline{D}_{\mathcal{Q}}(i)\}, \quad \text{otherwise} \end{cases} \tag{4.6}$$

$$M_{\mathcal{Q}}(i) = \max\{d_{ij} : j \in \mathcal{Q}(i)\} \tag{4.7}$$

$$\Delta_{\mathcal{Q}}(i) = \min\left\{\overline{D}_{\mathcal{Q}}(i) - \widetilde{D}_{\mathcal{Q}}(i), M_{\mathcal{Q}}(i)\right\}. \tag{4.8}$$

Given $k \in \mathcal{Q}$, let $\mathcal{Q}_k = \{1, 2, \ldots, k-1, k\}$ and $\mathcal{T}_{\mathcal{Q}_k} = \{\mathcal{V}_{\mathcal{Q}_k}, \mathcal{E}_{\mathcal{Q}_k}\}$ be the subtree of $\mathcal{T}$ with leaf nodes $\mathcal{Q}_k$. It is easily verified that, if $\mathcal{Q}$ satisfies (A1)-(A3) then every subset $\mathcal{Q}_k$ for $k = 1, \ldots, Q$ satisfies these properties as well.

Now, let $K \in \mathcal{Q}$, and suppose there exists a $j^* \in \mathcal{V}_{\mathcal{Q}_K}$ such that $j^* \in \mathcal{P}(K)$ and $\widetilde{D}_{\mathcal{Q}_K}(j^*) > 0$. Then there exists $r^* \in \mathcal{Q}$ such that $\widetilde{D}_{\mathcal{Q}_K}(j^*) = d_{0r^*}$. Clearly $1 \leq r^* \leq K$. Let $u^* = \text{argmax}\{t(i) : i \in \mathcal{V}_{\mathcal{Q}_{r^*}} \cap \mathcal{P}(K)\}$. Figure 2 illustrates the relative position of the nodes j^*, r^*, and u^*, and the set $\mathcal{V}_{\mathcal{Q}_{r^*}}$. In this figure $\mathcal{Q}_K = \{1, 2, 3, r^*, K-1, K\}$, $\mathcal{Q}_{r^*} = \{1, 2, 3, r^*\}$, $\mathcal{V}_{\mathcal{Q}_K}$ is the set of all nodes and $\mathcal{V}_{\mathcal{Q}_{r^*}}$ is the set of nodes within the dotted area as shown in the graph. From (A3), it follows that $u^* \in \mathcal{P}(r^*)$. If not, then there exists a $r' < r^*, r' \in \mathcal{Q}_{r^*}$ such that $u^* \in \mathcal{P}(r')$ since $u^* \in \mathcal{V}_{\mathcal{Q}_{r^*}}$. Thus, we have $r', K \in \mathcal{Q}(u^*)$ and

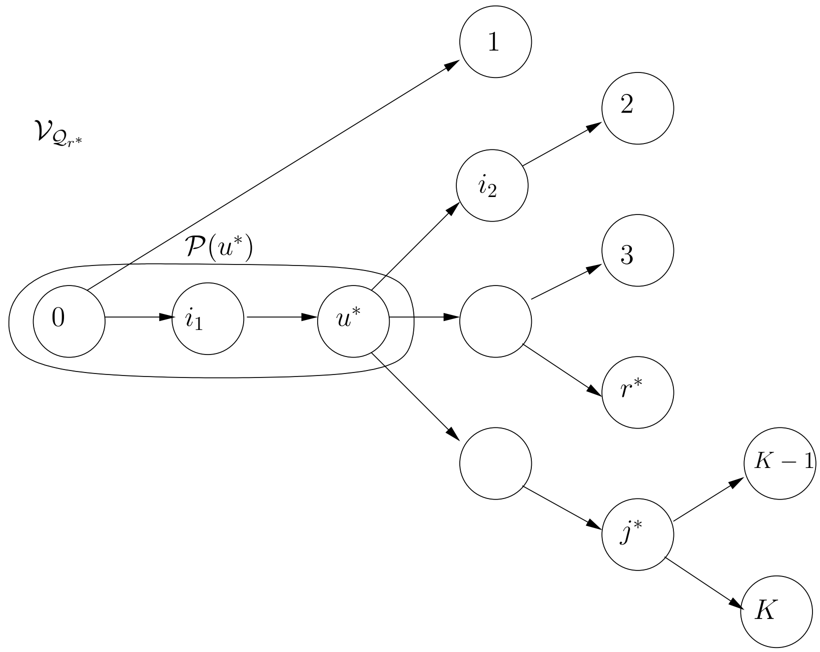

Figure 2: Notation for Lemmas 4.1 and 4.2

$r' < r^* \leq K$. Then according to (A3), we have $r^* \in \mathcal{Q}(u^*)$, which contradicts $u^* \notin \mathcal{P}(r^*)$. For K, j^*, r^* and u^* defined as above, we need the following two lemmas.

Lemma 4.1 *$\Delta_{\mathcal{Q}_K}(i) \geq \Delta_{\mathcal{Q}_{r^*}}(i)$ for all $i \in \mathcal{P}(u^*)$.*

Proof: We have

$$\overline{D}_{\mathcal{Q}_K}(i) = d_{0K} \geq d_{0r^*} = \overline{D}_{\mathcal{Q}_{r^*}}(i) \quad \text{for all } i \in \mathcal{P}(u^*). \tag{4.9}$$

Furthermore, for all $i \in \mathcal{P}(u^*)$, we have $r^*, K \in \mathcal{V}_{\mathcal{Q}_K}(i)$. It then follows from (A3) that $\mathcal{Q}_K(i) = \mathcal{Q}_{r^*}(i) \cup \{r^*+1, \ldots, K\}$. Thus

$$\begin{aligned} \mathcal{Q}_K \setminus \mathcal{Q}_K(i) &= \{1, \ldots, K\} \setminus (\mathcal{Q}_{r^*}(i) \cup \{r^*+1, \ldots, K\}) \\ &= (\{1, \ldots, K\} \setminus \{r^*+1, \ldots, K\}) \setminus \mathcal{Q}_{r^*}(i) \\ &= \mathcal{Q}_{r^*} \setminus \mathcal{Q}_{r^*}(i). \end{aligned} \tag{4.10}$$

(For example, in Figure 2, consider node $i_1 \in \mathcal{P}(u^*)$, then $Q_K(i_1) = \{2, 3, r^*, K-1, K\}$ and $\mathcal{Q}_{r^*}(i_1) = \{2, 3, r^*\}$. Thus $\mathcal{Q}_K \setminus \mathcal{Q}_K(i_1) = \mathcal{Q}_{r^*} \setminus \mathcal{Q}_{r^*}(i_1) = \{1\}$.)

Next, note that for all $i \in \mathcal{P}(u^*)$, (4.9) implies that $d_{0j} \leq \overline{D}_{\mathcal{Q}_K}(i) = d_{0K}$ for all $j \in \mathcal{Q}_K$ and $d_{0j} \leq \overline{D}_{\mathcal{Q}_{r^*}}(i) = d_{0r^*}$ for all $j \in \mathcal{Q}_{r^*}$. Thus for all nodes $i \in \mathcal{P}(u^*)$, $\widetilde{D}_{\mathcal{Q}_K}(i) = \max\{d_{0j}:$

$j \in \mathcal{Q}_K \setminus \mathcal{Q}_K(i)\}$ and $\widetilde{D}_{\mathcal{Q}_{r^*}}(i) = \max\{d_{0j} : j \in \mathcal{Q}_{r^*} \setminus \mathcal{Q}_{r^*}(i)\}$. It then follows from (4.10) that

$$\widetilde{D}_{\mathcal{Q}_K}(i) = \widetilde{D}_{\mathcal{Q}_{r^*}}(i) \quad \text{for all } i \in \mathcal{P}(u^*). \tag{4.11}$$

Since $\mathcal{Q}_{r^*}(i) \subset \mathcal{Q}_K(i)$, we also have

$$M_{\mathcal{Q}_K}(i) \geq M_{\mathcal{Q}_{r^*}}(i) \quad \text{for all } i \in \mathcal{P}(u^*). \tag{4.12}$$

The lemma follows from (4.9), (4.11), (4.12) and the definition of Δ. □

Lemma 4.2 *$\Delta_{\mathcal{Q}_K}(i) = \Delta_{\mathcal{Q}_{r^*}}(i)$ for all $i \in \mathcal{V}_{\mathcal{Q}_{r^*}} \setminus \mathcal{P}(u^*)$.*

Proof: We first claim that

$$j^* \notin \mathcal{V}_{\mathcal{Q}_{r^*}}. \tag{4.13}$$

Suppose that $j^* \in \mathcal{V}_{\mathcal{Q}_{r^*}}$. Then there exists $r_{j^*} \in \mathcal{Q}$ such that $r_{j^*} \leq r^* < K$, i.e., $r_{j^*} \in \mathcal{Q}_K(j^*)$. Note that by definition $r^* \notin \mathcal{Q}_K(j^*)$. Since $K \in \mathcal{Q}_K(j^*)$ and $r_{j^*} \leq r^* < K$, we have a contradiction to (A3). Thus (4.13) holds.

Next, we show that

$$\mathcal{Q}_{r^*}(i) = \mathcal{Q}_K(i) \quad \text{for all } i \in \mathcal{V}_{\mathcal{Q}_{r^*}} \setminus \mathcal{P}(u^*). \tag{4.14}$$

Clearly $\mathcal{Q}_{r^*}(i) \subseteq \mathcal{Q}_K(i)$. Now, suppose there exists a $k \in \mathcal{Q}_K(i)$ such that $k > r^*$. Note that $i \in \mathcal{V}_{\mathcal{Q}_{r^*}}$ and $j^* \notin \mathcal{V}_{\mathcal{Q}_{r^*}}$ from (4.13), thus $j^* \notin \mathcal{P}(i)$. Furthermore we also have $i \notin \mathcal{P}(j^*)$, otherwise by definition of u^* we would have $i \in \mathcal{P}(u^*)$. Thus $i \notin \mathcal{V}_{\mathcal{Q}_K}(j^*)$ and so $k \notin \mathcal{V}_{\mathcal{Q}_K}(j^*)$. Thus $d_{0r^*} = \widetilde{D}_{\mathcal{Q}_K}(j^*) = \max\{d_{0j} : j \in \mathcal{Q}_K \setminus \mathcal{Q}_K(j^*)$ and $d_{0j} \leq \overline{D}_{\mathcal{Q}_K}(j^*) = d_{0K}\} \geq d_{0k}$, which is a contradiction to $k > r^*$. Thus (4.14) is true. (The claim is clear in Figure 2. Consider the node $i_2 \in \mathcal{V}_{\mathcal{Q}_{r^*}} \setminus \mathcal{P}(u^*)$. Here $\mathcal{Q}_{r^*}(i_2) = \mathcal{Q}_K(i_2) = \{2\}$.)

From (4.14), we have

$$\overline{D}_{\mathcal{Q}_K}(i) = \overline{D}_{\mathcal{Q}_{r^*}}(i) \quad \text{for all } i \in \mathcal{V}_{\mathcal{Q}_{r^*}} \setminus \mathcal{P}(u^*), \tag{4.15}$$

and

$$M_{\mathcal{Q}_K}(i) = M_{\mathcal{Q}_{r^*}}(i) \quad \text{for all } i \in \mathcal{V}_{\mathcal{Q}_{r^*}} \setminus \mathcal{P}(u^*). \tag{4.16}$$

From (4.14) and (4.15), we have $\widetilde{D}_{\mathcal{Q}_K}(i) = \max\{d_{0j} : j \in \mathcal{Q}_K \setminus \mathcal{Q}_{r^*}(i) \text{ and } d_{0j} \leq \overline{D}_{\mathcal{Q}_{r^*}}(i)\}$. Now, consider the set

$$\begin{aligned}
& \{j : j \in \mathcal{Q}_K \setminus \mathcal{Q}_{r^*}(i) \text{ and } d_{0j} \leq \overline{D}_{\mathcal{Q}_{r^*}}(i)\} \\
= \; & \{j : j \in (\mathcal{Q}_{r^*} \cup \{r^*+1, \ldots, K\}) \setminus \mathcal{Q}_{r^*}(i) \text{ and } d_{0j} \leq \overline{D}_{\mathcal{Q}_{r^*}}(i)\} \\
= \; & \{j : j \in \mathcal{Q}_{r^*} \setminus \mathcal{Q}_{r^*}(i) \text{ and } d_{0j} \leq \overline{D}_{\mathcal{Q}_{r^*}}(i)\},
\end{aligned}$$

where the last step follows from the fact that $\overline{D}_{\mathcal{Q}_{r^*}}(i) \leq d_{0r^*}$ and $d_{0j} > d_{0r^*}$ for all $j \in \{r^*+1, \ldots, K\}$. Thus,

$$\widetilde{D}_{\mathcal{Q}_K}(i) = \widetilde{D}_{\mathcal{Q}_{r^*}}(i) \quad \text{for all } i \in \mathcal{V}_{\mathcal{Q}_{r^*}} \setminus \mathcal{P}(u^*). \tag{4.17}$$

The lemma follows from (4.15), (4.16), (4.17) and the definition of Δ. □

We are now ready to state the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequalities and prove their validity.

Theorem 4.1 *Given any $\mathcal{Q} \subseteq \mathcal{V}$ satisfying (A1), (A2), and (A3) and any subset $S_{\mathcal{Q}} \subseteq \mathcal{V}_{\mathcal{Q}}$, the inequality*

$$\sum_{i \in S_{\mathcal{Q}}} x_i + \sum_{i \in \overline{S}_{\mathcal{Q}}} \Delta_{\mathcal{Q}}(i) y_i \geq M_{\mathcal{Q}}(0),$$

where $\overline{S}_{\mathcal{Q}} = \mathcal{V}_{\mathcal{Q}} \setminus S_{\mathcal{Q}}$, called a $(\mathcal{Q}, S_{\mathcal{Q}})$ inequality, is valid for X_{SLS}.

Proof: We show by induction over $k \in \{1, \ldots, Q\}$ that any $(\mathcal{Q}_k, S_{\mathcal{Q}_k})$ inequality is valid for X_{SLS}.

The base case ($k = 1$): Note that $\overline{D}_{\mathcal{Q}_1}(i) = d_{01}$, $\widetilde{D}_{\mathcal{Q}_1}(i) = 0$, and $M_{\mathcal{Q}_1}(i) = d_{i1}$ for all $i \in \mathcal{V}_{Q_1}$. Given any point $(x, y) \in X_{\mathrm{SLS}}$, the left-hand-side of the $(\mathcal{Q}_1, S_{\mathcal{Q}_1})$ inequality is given by

$$\sum_{i \in S_{\mathcal{Q}_1}} x_i + \sum_{i \in \overline{S}_{\mathcal{Q}_1}} \min\{d_{01}, d_{i1}\} y_i = \sum_{i \in S_{\mathcal{Q}_1}} x_i + \sum_{i \in \overline{S}_{\mathcal{Q}_1}} d_{i1} y_i \geq d_{01} = M_{\mathcal{Q}_1}(0).$$

The first equality follows from the fact that $d_{01} \geq d_{i1}$, the inequality follows from the validity of the (ℓ, S) inequality with $\ell = 1$ and $S = S_{\mathcal{Q}_1}$, and the last equality follows from the definition of $M_{\mathcal{Q}_1}(0)$.

The inductive step: We assume that for all $k \in \{1, \ldots, K-1\}$ (where $K-1 < Q$), given any $S_{\mathcal{Q}_k} \subseteq \mathcal{V}_{\mathcal{Q}_k}$, the $(\mathcal{Q}_k, S_{\mathcal{Q}_k})$ inequality is valid for X_{SLS}. Consider any $S_{\mathcal{Q}_K} \subseteq \mathcal{V}_{\mathcal{Q}_K}$, we show that the $(\mathcal{Q}_K, S_{\mathcal{Q}_K})$ inequality

$$\sum_{i \in S_{\mathcal{Q}_K}} x_i + \sum_{i \in \overline{S}_{\mathcal{Q}_K}} \Delta_{\mathcal{Q}_K}(i) y_i \geq M_{\mathcal{Q}_K}(0)$$

is also valid for X_{SLS}.

Let $\mathcal{F}_K = \{i \in \mathcal{P}(K) \cap \overline{S}_{\mathcal{Q}_K} : \overline{D}_{\mathcal{Q}_K}(i) - \widetilde{D}_{\mathcal{Q}_K}(i) < M_{\mathcal{Q}_K}(i)\}$. Given any solution $(x, y) \in X_{\text{SLS}}$, we consider two cases: (a) there exists $j^* \in \mathcal{F}_K$ such that $y_{j^*} = 1$, and (b) $y_j = 0$ for all $j \in \mathcal{F}_K$.

Case (a): Note that $\overline{D}_{\mathcal{Q}_K}(j^*) - \widetilde{D}_{\mathcal{Q}_K}(j^*) < M_{\mathcal{Q}_K}(j^*)$ implies $\widetilde{D}_{\mathcal{Q}_K}(j^*) > 0$ since $\overline{D}_{\mathcal{Q}_K}(j^*) \geq M_{\mathcal{Q}_K}(j^*)$. Thus there exists $r^* \in \mathcal{Q}$ such that $\widetilde{D}_{\mathcal{Q}_K}(j^*) = d_{0r^*}$. Let $S_{\mathcal{Q}_{r^*}} = S_{\mathcal{Q}_K} \cap \mathcal{V}_{\mathcal{Q}_{r^*}}$ and $\overline{S}_{\mathcal{Q}_{r^*}} = \overline{S}_{\mathcal{Q}_K} \cap \mathcal{V}_{\mathcal{Q}_{r^*}}$. The left-hand-side of the $(\mathcal{Q}_K, S_{\mathcal{Q}_K})$ inequality is then equal to

$$\sum_{i \in S_{\mathcal{Q}_{r^*}}} x_i + \tag{4.18}$$

$$\sum_{i \in S_{\mathcal{Q}_K} \setminus S_{\mathcal{Q}_{r^*}}} x_i + \tag{4.19}$$

$$\sum_{i \in \overline{S}_{\mathcal{Q}_{r^*}}} \Delta_{\mathcal{Q}_K}(i) y_i + \tag{4.20}$$

$$\sum_{i \in \overline{S}_{\mathcal{Q}_K} \setminus \overline{S}_{\mathcal{Q}_{r^*}}} \Delta_{\mathcal{Q}_K}(i) y_i. \tag{4.21}$$

As before, let $u^* = \text{argmax}\{t(i) : i \in \mathcal{V}_{\mathcal{Q}_{r^*}} \cap \mathcal{P}(K)\}$. Expression (4.20) can be further disaggregated into

$$\sum_{i \in \overline{S}_{\mathcal{Q}_{r^*}} \cap \mathcal{P}(u^*)} \Delta_{\mathcal{Q}_K}(i) y_i + \tag{4.22}$$

$$\sum_{i \in \overline{S}_{\mathcal{Q}_{r^*}} \setminus \mathcal{P}(u^*)} \Delta_{\mathcal{Q}_K}(i) y_i. \tag{4.23}$$

From Lemma 4.1, it follows that

$$(4.22) \geq \sum_{i \in \overline{S}_{\mathcal{Q}_{r^*}} \cap \mathcal{P}(u^*)} \Delta_{\mathcal{Q}_{r^*}}(i) y_i,$$

and from Lemma 4.2, it follows that

$$(4.23) = \sum_{i \in \overline{S}_{\mathcal{Q}_{r^*}} \setminus \mathcal{P}(u^*)} \Delta_{\mathcal{Q}_{r^*}} y_i.$$

From the validity of the $(\mathcal{Q}_{r^*}, S_{\mathcal{Q}_{r^*}})$ inequality, we then have

$$(4.18) + (4.22) + (4.23) \geq M_{\mathcal{Q}_{r^*}}(0) = d_{0r^*}.$$

Now consider the expression (4.21). Since $j^* \in \overline{S}_{\mathcal{Q}_K} \setminus \overline{S}_{\mathcal{Q}_{r^*}}$ and all coefficients are non-negative, we have that

$$(4.21) \geq \overline{D}_{\mathcal{Q}_K}(j^*) - \widetilde{D}_{\mathcal{Q}_K}(j^*) = d_{0K} - d_{0r^*}.$$

Thus

$$(4.18) + (4.22) + (4.23) + (4.21) \geq d_{0K},$$

which implies

$$(4.18) + (4.19) + (4.22) + (4.23) + (4.21) \geq d_{0K} = M_{\mathcal{Q}_K}(0).$$

Therefore the $(\mathcal{Q}_K, S_{\mathcal{Q}_K})$ inequality is valid.

Case (b): The left-hand-side of the $(\mathcal{Q}_K, S_{\mathcal{Q}_K})$ inequality equals

$$\begin{aligned}
& \sum_{i \in S_{\mathcal{Q}_K}} x_i + \sum_{i \in \overline{S}_{\mathcal{Q}_K}} \Delta_{\mathcal{Q}_K}(i) y_i \\
\geq \quad & \sum_{i \in S_{\mathcal{Q}_K} \cap \mathcal{P}(K)} x_i + \sum_{i \in \overline{S}_{\mathcal{Q}_K} \cap \mathcal{P}(K)} \Delta_{\mathcal{Q}_K}(i) y_i \\
= \quad & \sum_{i \in S_{\mathcal{Q}_K} \cap \mathcal{P}(K)} x_i + \sum_{i \in \overline{S}_{\mathcal{Q}_K} \cap \mathcal{P}(K)} M_{\mathcal{Q}_K}(i) y_i \\
= \quad & \sum_{i \in S_{\mathcal{Q}_K} \cap \mathcal{P}(K)} x_i + \sum_{i \in \overline{S}_{\mathcal{Q}_K} \cap \mathcal{P}(K)} d_{iK} y_i \\
\geq \quad & d_{0K} = M_{\mathcal{Q}_{K(0)}},
\end{aligned}$$

where the third expression follows from the fact that $y_j = 0$ for all $j \in \overline{S}_{\mathcal{Q}_K} \cap \mathcal{P}(K)$ such that $\overline{D}_{\mathcal{Q}_K}(j) - \widetilde{D}_{\mathcal{Q}_K}(j) < M_{\mathcal{Q}_K}(j)$, the fourth expression follows from the definition of $M_{\mathcal{Q}_K}(j)$, and the fifth expression follows from the validity of the (ℓ, S) inequality with $\ell = K$ and

$S = S_{\mathcal{Q}_K} \cap \mathcal{P}(K)$. Therefore the $(\mathcal{Q}_K, S_{\mathcal{Q}_K})$ inequality is valid. $\square$

We conclude this section with a discussion of properties (A1) and (A2) and an example that illustrates the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequalities. Suppose property (A1) does not hold for some $\mathcal{Q}$. In particular, suppose there exists a pair of nodes $q_1, q_2 \in \mathcal{Q}$ such that $d_{0q_1} = d_{0q_2}$. Without loss of generality, we index the nodes in $\mathcal{Q}$ such that $q_2 > q_1$. Let $\mathcal{Q}' = \mathcal{Q} \setminus \{q_2\}$. Note that $\mathcal{Q}'$ satisfies (A1). From the fact that $d_{0q_1} = d_{0q_2}$, it can be easily verified that $\Delta_{\mathcal{Q}'}(i) = \Delta_{\mathcal{Q}}(i)$ for all $i \in \mathcal{V}_{\mathcal{Q}'}$ and $M_{\mathcal{Q}'}(0) = M_{\mathcal{Q}}(0)$. Now, let $S_{\mathcal{Q}'} = S_{\mathcal{Q}} \cap \mathcal{V}_{\mathcal{Q}'}$ and $\overline{S}_{\mathcal{Q}'} = \overline{S}_{\mathcal{Q}} \cap \mathcal{V}_{\mathcal{Q}'}$. Then

$$\begin{aligned}
&\sum_{i \in S_{\mathcal{Q}}} x_i + \sum_{i \in \overline{S}_{\mathcal{Q}}} \Delta_{\mathcal{Q}}(i) y_i \\
&\geq \sum_{i \in S_{\mathcal{Q}'}} x_i + \sum_{i \in \overline{S}_{\mathcal{Q}'}} \Delta_{\mathcal{Q}}(i) y_i \\
&= \sum_{i \in S_{\mathcal{Q}'}} x_i + \sum_{i \in \overline{S}_{\mathcal{Q}'}} \Delta_{\mathcal{Q}'}(i) y_i \\
&\geq M_{\mathcal{Q}'}(0) = M_{\mathcal{Q}}(0).
\end{aligned}$$

Thus the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequality is valid. However, this inequality is clearly dominated by the $(\mathcal{Q}', S_{\mathcal{Q}'})$ inequality. Consequently, (A1) is without loss of generality.

Suppose property (A2) does not hold for some $\mathcal{Q}$ and there exists a pair of nodes $q_1, q_2 \in \mathcal{Q}$ such that $q_1 \in \mathcal{P}(q_2)$. Then $\mathcal{V}_{\mathcal{Q}} = \mathcal{V}_{\mathcal{Q} \setminus \{q_1\}}$ and we only need to consider $(\mathcal{Q}, S_{\mathcal{Q}})$ inequalities corresponding to $\mathcal{Q} \setminus \{q_1\}$ instead of $\mathcal{Q}$. Consequently, (A2) is without loss of generality.

Example: Consider an instance of (SLS) with 7 nodes as shown in Figure 3. The problem parameters are shown in the columns labelled $\overline{\alpha}_i$, $\overline{\beta}_i$ and d_i in Table 1. The optimal LP relaxation objective value of (SLS) is 3011.84 and the corresponding optimal solution (x, y) is shown in the columns labelled x^1 and y^1 in Table 1. We augment the LP

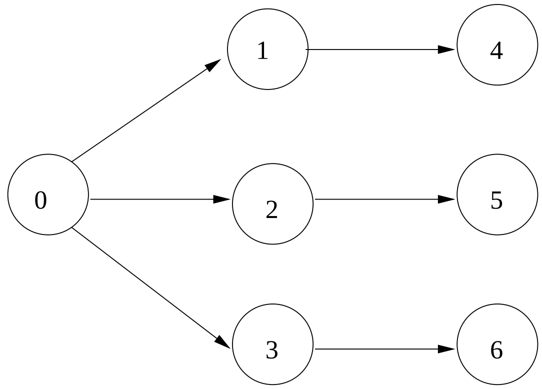

Figure 3: The scenario tree for the example

relaxation with 3 $(\mathcal{Q}, S_{\mathcal{Q}})$ inequalities:

$$10y_0 \geq 10, \quad \text{i.e.,} \quad \mathcal{Q} = \{0\}, \overline{S}_{\mathcal{Q}} = \{0\}$$

$$x_0 + x_2 + 5y_3 \geq 35, \quad \text{i.e.,} \quad \mathcal{Q} = \{2,3\}, \overline{S}_{\mathcal{Q}} = \{3\}$$

$$x_0 + 5y_1 + 20y_2 + 5y_4 \geq 35, \quad \text{i.e.,} \quad \mathcal{Q} = \{2,4\}, \overline{S}_{\mathcal{Q}} = \{1,2,4\}$$

Then we obtain an integral optimal solution as shown in columns labelled x^2 and y^2 in Table 1 and the corresponding optimal objective value is 3143.

Table 1: Data for the example

	$\overline{\alpha}_i$	$\overline{\beta}_i$	d_i	x^1	y^1	x^2	y^2
0	100	1	10	25	0.56	30	1
1	10	2000	15	0	0.00	0	0
2	10	2000	20	5	0.14	0	0
3	10	30	25	10	0.29	5	1
4	1	30	10	10	1.00	5	1
5	1	1	15	15	1.00	15	1
6	1	1	10	10	1.00	10	1

4.3 Facets for the Stochastic Lot-Sizing Problem

In this section, we give some classes of facets for the stochastic lot-sizing polyhedron. First, we identify some facets from the original inequalities defining X_{SLS}. Next, we provide necessary and sufficient conditions under which a $(\mathcal{Q}, S_{\mathcal{Q}})$ inequality is facet-defining.

Throughout the remainder of this chapter, we make the assumption

(A4) $d_i > 0$ for all $i \in \mathcal{V}$.

Given (A4), the following results can be shown by constructing appropriate sets of affinely independent solutions. Recall that $|\mathcal{V}| = N$.

Proposition 4.1 *The dimension of* X_{SLS} *is* $2N - 1$.

Proposition 4.2 *The inequalities*

(i) $x_i \leq M_i y_i$ *for* $i \in \mathcal{V} \setminus \{0\}$,

(ii) $y_i \leq 1$ *for* $i \in \mathcal{V} \setminus \{0\}$,

(iii) $x_i \geq 0$ *for* $i \in \mathcal{V} \setminus \{0\}$,

are facet-defining for X_{SLS}.

Note that, the inequalities $y_i \geq 0$, $i \in \mathcal{V} \setminus \{0\}$, are not facet-defining because $y_i = 0$ implies $x_i = 0$, and therefore we can have no more than $2N-2$ affinely independent solutions satisfying $y_i = 0$.

We now establish a set of conditions guaranteeing that a $(\mathcal{Q}, S_{\mathcal{Q}})$ inequality is facet-defining. Let $\mathcal{F}_{\mathcal{Q}} = \{i \in \overline{S}_{\mathcal{Q}} : \overline{D}_{\mathcal{Q}}(i) - \widetilde{D}_{\mathcal{Q}}(i) < M_{\mathcal{Q}}(i)\}$ and $\mathcal{G}_{\mathcal{Q}} = \overline{S}_{\mathcal{Q}} \setminus \mathcal{F}_{\mathcal{Q}}$. Thus, $\mathcal{V}_{\mathcal{Q}} = \mathcal{F}_{\mathcal{Q}} \cup \mathcal{G}_{\mathcal{Q}} \cup S_{\mathcal{Q}}$. We need the following definitions.

Definition 4.1 *Given* $\mathcal{Q} \subseteq \mathcal{V}$ *and* $S_{\mathcal{Q}} \subseteq \mathcal{V}_{\mathcal{Q}}$, *the* neighborhood *of* $(\mathcal{Q}, S_{\mathcal{Q}})$ *is*

$$\mathcal{N}(\mathcal{Q}, S_{\mathcal{Q}}) = \bigcup_{i \in \mathcal{V}_{\mathcal{Q}} \setminus (\cup_{j \in \overline{S}_{\mathcal{Q}}} \mathcal{V}_{\mathcal{Q}}(j))} \mathcal{C}(i) \setminus \mathcal{V}_{\mathcal{Q}}.$$

For example, in Figure 4, let $\mathcal{Q} = \{1, 2, 3, 4\}$ and $S_{\mathcal{Q}} = \{0, 3, 5, 9\}$, then $\mathcal{N}(\mathcal{Q}, S_{\mathcal{Q}})$ contains the two nodes shaded horizontally.

Definition 4.2 *Given* $j \in \mathcal{V}_{\mathcal{Q}}$, *let* $q_j = \max\{i : \ i \in \mathcal{Q}(j)\}$ *and*

$$\mathcal{W}(j) = \bigcup_{i \in \mathcal{Q} \setminus \mathcal{Q}_{q_j}} \operatorname{argmin} \left\{ t(m) : m \in \overline{S}_{\mathcal{Q}} \cap \mathcal{P}(i) \setminus \mathcal{V}_{\mathcal{Q}_{q_j}} \right\}.$$

For example, in Figure 4, if $j = 9$ then $q_j = 2$ and $\mathcal{W}(j) = \{4, 7\}$; and if $j = 6$ then $q_j = 3$ and $\mathcal{W}(j) = \{4\}$.

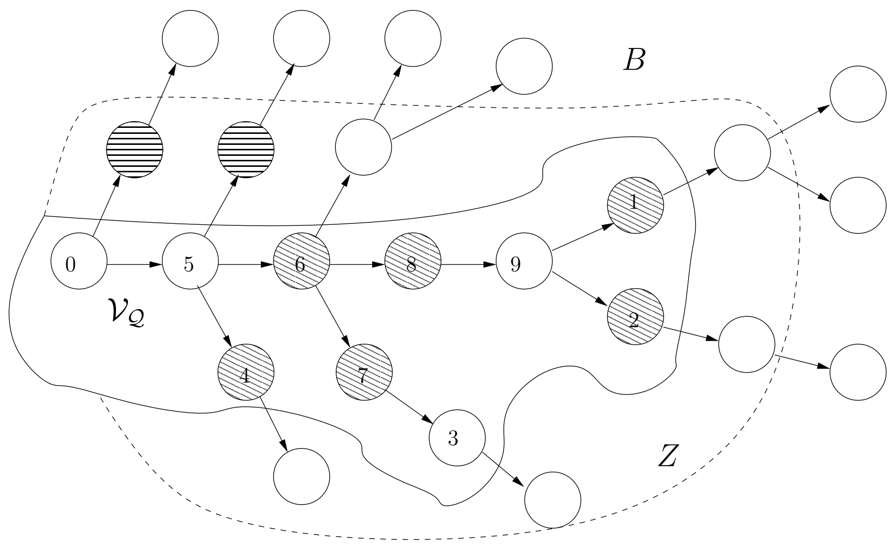

Figure 4: Partitioning of the node set $\mathcal{V}$ used in the proof of Theorem 4.2

Theorem 4.2 *The* $(\mathcal{Q}, S_{\mathcal{Q}})$ *inequality*

$$\sum_{i \in S_{\mathcal{Q}}} x_i + \sum_{i \in \overline{S}_{\mathcal{Q}}} \Delta_{\mathcal{Q}}(i) y_i \geq M_{\mathcal{Q}}(0)$$

is facet-defining if and only if

(i) $0 \in S_{\mathcal{Q}}$,

(ii) $M_{\mathcal{Q}}(0) \geq \max_{i \in \mathcal{N}(\mathcal{Q}, S_{\mathcal{Q}})}\{d_{0i}\}$,

(iii) *For each* $j \in \mathcal{V}_{\mathcal{Q}}$,

> *(a)* $\mathcal{W}(j) \cap \mathcal{P}(i) \neq \emptyset, \ \forall i \in \mathcal{Q} \setminus \mathcal{Q}_{q_j}$,
>
> *(b) If* $j \in \mathcal{F}_{\mathcal{Q}}$, *then* $\widetilde{D}_{\mathcal{Q}}(j) \geq d_{0a(k)}, \ \forall k \in \mathcal{W}(j)$,
>
> *(c) If* $j \in \mathcal{G}_{\mathcal{Q}}$, *then* $d_{0a(j)} \geq d_{0a(k)}, \ \forall k \in \mathcal{W}(j)$,
>
> *(d) If* $j \in S_{\mathcal{Q}}$, *then* $\overline{D}_{\mathcal{Q}}(j) > d_{0a(k)}, \ \forall k \in \mathcal{W}(j)$,

(iv) $(\cup_{i \in \mathcal{G}_{\mathcal{Q}}} \mathrm{argmax}\{j : j \in \mathcal{Q}(i)\}) \cap \mathcal{L} = \emptyset$.

Proof: Proof of sufficiency.

We first describe the construction of $2N-1$ vectors that are in X_{SLS} and satisfy the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequality at equality. Then we show that the vectors are linearly independent.

Given the $(\mathcal{Q}, S_\mathcal{Q})$ inequality, we partition $\mathcal{V}$ into disjoint sets $\mathcal{V} = \{0\} \cup A \cup Z \cup B$, where $A = \mathcal{V}_\mathcal{Q} \setminus \{0\}$, $Z = \{j : j \in \mathcal{V} \setminus \mathcal{V}_\mathcal{Q} \text{ and } a(j) \in \mathcal{V}_\mathcal{Q}\}$ and $B = \mathcal{V} \setminus (\mathcal{V}_\mathcal{Q} \cup Z)$. Note that we have $\mathcal{N}(\mathcal{Q}, S_\mathcal{Q}) \subseteq Z$. Nodes in the set $\mathcal{V} \setminus \mathcal{V}_\mathcal{Q}$ correspond to a forest, and Z represents the set of root nodes of the subtrees in this forest. This partitioning is illustrated in Figure 4. Here $\mathcal{Q} = \{1, 2, 3, 4\}$, $\mathcal{V}_\mathcal{Q} = \{0, 1, 2, \ldots, 9\}$, $S_\mathcal{Q} = \{0, 3, 5, 9\}$, $\overline{S}_\mathcal{Q} = \{1, 2, 4, 6, 7, 8\}$ (shaded diagonally), and $A = \{1, 2, \ldots, 9\}$. The two horizontally shaded nodes in Z represent $\mathcal{N}(\mathcal{Q}, S_\mathcal{Q})$.

Construction: We create one vector u^0 for the root node $\{0\}$ and two vectors u^j and v^j for each node $j \in \mathcal{V} \setminus \{0\}$.

We let

$$u^0 = M_\mathcal{Q}(0)e^{x_0} + e^{y_0} + \sum_{i \in Z}(M_i e^{x_i} + e^{y_i}),$$

where e^{x_i} and e^{y_i} are unit vectors in $\mathbb{R}^{2N}$ corresponding to the coordinates x_i and y_i, respectively.

$j \in B$: We let

$$\begin{aligned} u^j = &\quad u^0 + e^{y_j}, \quad \text{and} \\ v^j = &\quad u^0 + M_j e^{x_j} + e^{y_j}. \end{aligned}$$

$j \in A$:

If $j \in S_\mathcal{Q}$, we let

$$\begin{aligned} u^j = &\quad u^0 + (\overline{D}_\mathcal{Q}(j) - \varepsilon - M_\mathcal{Q}(0))e^{x_0} \\ &\qquad + \varepsilon e^{x_j} + e^{y_j} \\ &\qquad + \textstyle\sum_{i \in \mathcal{W}(j)}(M_\mathcal{Q}(i)e^{x_i} + e^{y_i}), \end{aligned}$$

where ε is a sufficiently small positive number, and

$$v^j = \quad u^0 + e^{y_j}.$$

If $j \in \overline{S}_\mathcal{Q}$, we let

$$\begin{aligned} u^j = &\quad u^0 + (\overline{D}_\mathcal{Q}(j) - \Delta_\mathcal{Q}(j) - M_\mathcal{Q}(0))e^{x_0} \\ &\qquad + \Delta_\mathcal{Q}(j)e^{x_j} + e^{y_j} \\ &\qquad + \textstyle\sum_{i \in \mathcal{W}(j)}(M_\mathcal{Q}(i)e^{x_i} + e^{y_i}) \quad \text{and} \\ v^j = &\quad u^j + \varepsilon e^{x_j}. \end{aligned}$$

$j \in Z$:

If $j \in \mathcal{N}(\mathcal{Q}, S_{\mathcal{Q}})$, we let

$$\begin{aligned} u^j = &\ u^0 - M_j e^{x_j} - e^{y_j} + \textstyle\sum_{i\in B}(M_i e^{x_i} + e^{y_i}) \quad \text{and} \\ v^j = &\ u^j + e^{y_j}. \end{aligned}$$

If $j \in Z \setminus \mathcal{N}(\mathcal{Q}, S_{\mathcal{Q}})$, define $k_j = \operatorname{argmin}\{t(i) : \ i \in \overline{S}_{\mathcal{Q}} \cap \mathcal{P}(j)\}$. Note that $k_j \in \overline{S}_{\mathcal{Q}}$ by definition. We let

$$\begin{aligned} u^j = &\ u^{k_j} + (M_{k_j} - \Delta_{\mathcal{Q}}(k_j))e^{x_{k_j}} - M_j e^{x_j} - e^{y_j} \quad \text{and} \\ v^j = &\ u^j + e^{y_j}. \end{aligned}$$

Feasibility: It is obvious that $u^0 \in X_{SLS}$. Consequently, the vectors $\{u^j, v^j\}_{j\in B}$ and $\{v^j\}_{j\in S_{\mathcal{Q}}}$ are also feasible.

Now we verify the feasibility of u^j for $j \in \overline{S}_{\mathcal{Q}}$. Given $j \in \overline{S}_{\mathcal{Q}}$, u^j satisfies $0 \le x_i \le M_i y_i$ and $y_i \in \{0,1\}$ for all $i \in \mathcal{V}$ since $x_0 < M_{\mathcal{Q}}(0) \le M_0$, $\Delta_{\mathcal{Q}}(j) \le M_{\mathcal{Q}}(j) \le M_j$ and $M_{\mathcal{Q}}(k) \le M_k \ \forall k \in \mathcal{W}(j)$. Therefore, we just need to check that u^j satisfies constraint (4.2) for all $i \in \mathcal{V} = \{0\} \cup A \cup Z \cup B$.

Clearly u^j satisfies constraint (4.2) for $i = 0$. Also, note that if u^j satisfies constraint (4.2) for $i \in \{0\} \cup A$, then it satisfies constraint (4.2) for $i \in Z \cup B$ since $x_i = M_i$ and $y_i = 1$ for all $i \in Z$, and the nodes in Z include an ancestor of each node in B. Therefore, we just need to show that u^j satisfies constraint (4.2) for $i \in A = S_{\mathcal{Q}} \cup \overline{S}_{\mathcal{Q}}$.

Note that u^j yields

$$\begin{aligned} x_0 &= \overline{D}_{\mathcal{Q}}(j) - \Delta_{\mathcal{Q}}(j) \\ &\ge \overline{D}_{\mathcal{Q}}(j) - M_{\mathcal{Q}}(j) \\ &= d_{0a(j)}, \end{aligned} \tag{4.24}$$

where the second line follows from the definition of $\Delta_{\mathcal{Q}}(j)$ and the third line follows from the definition of $\overline{D}_{\mathcal{Q}}(j)$ and $M_{\mathcal{Q}}(j)$. It then follows that u^j satisfies constraint (4.2) for all $i \in \mathcal{P}(a(j))$.

Next, note that u^j yields

$$\begin{aligned} x_0 &= \overline{D}_{\mathcal{Q}}(j) - \Delta_{\mathcal{Q}}(j) \\ &\geq \overline{D}_{\mathcal{Q}}(j) - (\overline{D}_{\mathcal{Q}}(j) - \widetilde{D}_{\mathcal{Q}}(j)) \\ &= \widetilde{D}_{\mathcal{Q}}(j), \end{aligned} \tag{4.25}$$

where the second line follows from the definition of $\Delta_{\mathcal{Q}}(j)$. If $\widetilde{D}_{\mathcal{Q}}(j) > 0$, then we know that there exists $r_j \in Q$ such that $\widetilde{D}_{\mathcal{Q}}(j) = d_{0r_j}$. Thus (4.25) implies that u^j satisfies constraint (4.2) for all $i \in \mathcal{V}_{\mathcal{Q}_{r_j}}$.

Also, note that u^j yields

$$x_0 + x_j = \overline{D}_{\mathcal{Q}}(j). \tag{4.26}$$

Since $0 \in \mathcal{P}(i)$ and $j \in \mathcal{P}(i)$ for all $i \in \mathcal{V}_Q(j)$, (4.26) implies that u^j satisfies (4.2) for all $i \in \mathcal{V}_Q(j)$.

Next, considering (b) and (c) of condition (iii), (4.24) and (4.25) imply that u^j satisfies

$$x_0 \geq d_{0a(k)} \quad \forall k \in \mathcal{W}(j). \tag{4.27}$$

Thus u^j satisfies (4.2) for all $i \in \mathcal{P}(a(k)) \;\; \forall k \in \mathcal{W}(j)$.

Finally, note that

$$\{0\} \cup A = \mathcal{V}_{\mathcal{Q}} = \mathcal{P}(j) \cup \mathcal{V}_{\mathcal{Q}_{r_j}} \cup \mathcal{V}_Q(j) \cup \left(\bigcup_{k \in \mathcal{W}(j)} \mathcal{P}(a(k)) \right) \cup \left(\bigcup_{k \in \mathcal{W}(j)} \mathcal{V}_Q(k) \right).$$

So it only remains to check that u^j satisfies (4.2) for all $i \in \bigcup_{k \in \mathcal{W}(j)} \mathcal{V}_Q(k)$. Given any $k \in \mathcal{W}(j)$, note that u^j satisfies

$$\begin{aligned} x_0 + x_k &\geq d_{0a(k)} + M_{\mathcal{Q}}(k) \\ &= \overline{D}_{\mathcal{Q}}(k), \end{aligned} \tag{4.28}$$

where the first line follows from (4.27) and the second line follows from the definition of $\overline{D}_{\mathcal{Q}}(k)$. Since, for all $i \in \mathcal{V}(k)$ we have $0 \in \mathcal{P}(i)$, $k \in \mathcal{P}(i)$ and $d_{0i} \leq \overline{D}_{\mathcal{Q}}(k)$, it follows that u^j satisfies constraint (4.2) for all $i \in \mathcal{V}_Q(k)$ and $k \in \mathcal{W}(j)$.

v^j for $j \in \overline{S}_{\mathcal{Q}}$ is feasible because v^j satisfies constraint (4.2) since $v^j \geq u^j$ and condition (iv) ensures that v^j satisfies $0 \leq x_i \leq M_i y_i$ and $y_i \in \{0, 1\}$.

The feasibility of u^j for $j \in S_{\mathcal{Q}}$ can be established using analogous arguments as long as $\varepsilon \leq \Delta_{\mathcal{Q}}(j)$, $\overline{D}_{\mathcal{Q}}(j) - \varepsilon \geq \widetilde{D}_{\mathcal{Q}}(j)$ and $\overline{D}_{\mathcal{Q}}(j) - \varepsilon \geq d_{0a(k)} \ \ \forall k \in \mathcal{W}(j)$.

We now verify the feasibility of u^j for $j \in \mathcal{N}(\mathcal{Q}, S_{\mathcal{Q}})$. As before, we only need to verify that u^j satisfies constraint (4.2) for all $i \in \mathcal{V}$. Since the construction of u^j only affects nodes $i \in \mathcal{V}(j)$, from the feasibility of u^0, constraint (4.2) is satisfied for all $i \in \mathcal{V} \setminus \mathcal{V}(j)$. Given any node $i \in \mathcal{V}(j)$, note that u^j satisfies

$$\begin{aligned} \textstyle\sum_{k \in \mathcal{P}(i)} x_k &= M_{\mathcal{Q}}(0) + \textstyle\sum_{k \in \mathcal{P}(i) \setminus \mathcal{P}(j)} M_k \\ &\geq d_{0j} + \textstyle\sum_{k \in \mathcal{P}(i) \setminus \mathcal{P}(j)} d_k \\ &= d_{0i}, \end{aligned} \tag{4.29}$$

where the first line follows from the construction of u^j and the second line follows from condition (ii). Thus u^j satisfies (4.2) for all $i \in \mathcal{V}$.

We now verify the feasibility of u^j for $j \in Z \setminus \mathcal{N}(\mathcal{Q}, S_{\mathcal{Q}})$. Since the construction of u^j only affects nodes $i \in \mathcal{V}(k_j)$, from the feasibility of u^{k_j} (recall that $k_j \in \overline{S}_{\mathcal{Q}}$), constraint (4.2) is satisfied for all $i \in \mathcal{V} \setminus \mathcal{V}(k_j)$. Given any node $i \in \mathcal{V}(k_j)$, note that u^j satisfies

$$\begin{aligned} x_0 + x_{k_j} &\geq d_{0a(k_j)} + M_{k_j} \\ &\geq d_{0i}, \end{aligned} \tag{4.30}$$

where the first line follows from (4.27) and the construction of u^j, and the second line follows the definition of M_{k_j} and the fact that $k_j \in \mathcal{P}(i)$. Thus u^j satisfies constraint (4.2) for all $i \in \mathcal{V}$.

Finally, v^j for $j \in Z$ is feasible since $v^j \geq u^j$.

Tightness of the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequality: Here we prove the claim that the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequality is tight or active at each of the solutions vectors u^0 and $\{u^j, v^j\}_{j \in \mathcal{V} \setminus \{0\}}$. This claim is true for u^0, $\{u^j, v^j\}_{j \in B}$, $\{v^j\}_{j \in S_{\mathcal{Q}}}$ and $\{u^j, v^j\}_{j \in \mathcal{N}(\mathcal{Q}, S_{\mathcal{Q}})}$. Furthermore, for any $j \in Z \setminus \mathcal{N}(\mathcal{Q}, S_{\mathcal{Q}})$, we have that u^j and v^j satisfy the claim as long as u^{k_j} satisfies the claim since $k_j \in \overline{S}_{\mathcal{Q}}$. Similarly the solutions $\{v^j\}_{j \in \overline{S}_{\mathcal{Q}}}$ satisfy the claim as long as $\{u^j\}_{j \in \overline{S}_{\mathcal{Q}}}$ satisfy the claim. Therefore, we just need to prove the claim for $\{u^j\}_{j \in S_{\mathcal{Q}} \cup \overline{S}_{\mathcal{Q}}}$.

For the case $\{u^j\}_{j\in\overline{S}_{\mathcal{Q}}}$, since $j \in \overline{S}_{\mathcal{Q}}$ and $\mathcal{W}(j) \subseteq \overline{S}_{\mathcal{Q}}$, we have that u^j satisfies

$$x_i^j = \begin{cases} \overline{D}_{\mathcal{Q}}(j) - \Delta_{\mathcal{Q}}(j) & \text{if } i = 0 \\ 0 & \text{if } i \in S_{\mathcal{Q}} \setminus \{0\}, \end{cases}$$

and

$$y_i^j = \begin{cases} 1 & \text{if } i \in \{j\} \cup \mathcal{W}(j) \\ 0 & \text{if } i \in \overline{S}_{\mathcal{Q}} \setminus (\{j\} \cup \mathcal{W}(j)). \end{cases}$$

For the case $\{u^j\}_{j\in S_{\mathcal{Q}}}$, since $j \in S_{\mathcal{Q}}$ and $\mathcal{W}(j) \subseteq \overline{S}_{\mathcal{Q}}$, we have that u^j satisfies

$$x_i^j = \begin{cases} \overline{D}_{\mathcal{Q}}(j) - \varepsilon & \text{if } i = 0 \\ 0 & \text{if } i \in S_{\mathcal{Q}} \setminus (\{0\} \cup \{j\}) \\ \varepsilon & \text{if } i = j, \end{cases}$$

and

$$y_i^j = \begin{cases} 1 & \text{if } i \in \mathcal{W}(j) \\ 0 & \text{if } i \in \overline{S}_{\mathcal{Q}} \setminus \mathcal{W}(j). \end{cases}$$

Thus u^j satisfies

$$\sum_{i\in S_{\mathcal{Q}}} x_i^j + \sum_{i\in\overline{S}_{\mathcal{Q}}} \Delta_{\mathcal{Q}}(i) y_i^j = \overline{D}_{\mathcal{Q}}(j) + \sum_{i\in\mathcal{W}(j)} \Delta_{\mathcal{Q}}(i) \tag{4.31}$$

for both cases that $\{u^j\}_{j\in S_{\mathcal{Q}}}$ and $\{u^j\}_{j\in\overline{S}_{\mathcal{Q}}}$.

It remains to show that the right-hand side of the above expression is equal to $M_{\mathcal{Q}}(0)$.

If $\mathcal{W}(j) = \emptyset$ then $\overline{D}_{\mathcal{Q}}(j) = M_{\mathcal{Q}}(0)$ by definition of $\mathcal{W}(j)$. If $\mathcal{W}(j) \neq \emptyset$, note that for all $i \in \mathcal{W}(j)$,

$$\overline{D}_{\mathcal{Q}}(i) - \widetilde{D}_{\mathcal{Q}}(i) \leq \overline{D}_{\mathcal{Q}}(i) - \overline{D}_{\mathcal{Q}}(j) \leq \overline{D}_{\mathcal{Q}}(i) - d_{0a(i)} = M_{\mathcal{Q}}(i),$$

where the first inequality follows from the fact that $\widetilde{D}_{\mathcal{Q}}(i) \geq d_{0q_j} = \overline{D}_{\mathcal{Q}}(j)$ since $i \notin \mathcal{V}_{\mathcal{Q}_{q_j}}$, and the second inequality follows from the fact that $\overline{D}_{\mathcal{Q}}(j) \geq \widetilde{D}_{\mathcal{Q}}(j) \geq d_{0a(i)}$ from case (b) of condition (iii) or $\overline{D}_{\mathcal{Q}}(j) \geq d_{0a(j)} \geq d_{0a(i)}$ from case (c) of condition (iii). Thus, for any node $i \in \mathcal{W}(j)$,

$$\Delta_{\mathcal{Q}}(i) = \overline{D}_{\mathcal{Q}}(i) - \widetilde{D}_{\mathcal{Q}}(i). \tag{4.32}$$

By Property (A2), we index the nodes in $\mathcal{W}(j)$ as $i_1, i_2, \ldots, i_W$ such that $\overline{D}_{\mathcal{Q}}(i_1) < \overline{D}_{\mathcal{Q}}(i_2) < \ldots < \overline{D}_{\mathcal{Q}}(i_W)$. From this indexing scheme, the definition of $\overline{D}_{\mathcal{Q}}$, $\widetilde{D}_{\mathcal{Q}}$, and $\mathcal{W}(j)$, it follows that $\widetilde{D}_{\mathcal{Q}}(i_1) = \overline{D}_{\mathcal{Q}}(j)$, $\overline{D}_{\mathcal{Q}}(i_W) = M_{\mathcal{Q}}(0)$, and

$$\overline{D}_{\mathcal{Q}}(i_k) = \widetilde{D}_{\mathcal{Q}}(i_{k+1}) \qquad k = 1, 2, \ldots, W-1.$$

Thus

$$\overline{D}_{\mathcal{Q}}(j) + \sum_{i \in \mathcal{W}(j)} \Delta_{\mathcal{Q}}(i) = M_Q(0),$$

and the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequality is tight for u^j for each $j \in \overline{S}_{\mathcal{Q}} \cup S_{\mathcal{Q}}$.

Linear Independence: Given the $2N-1$ vectors u^0 and $\{u^j, v^j\}_{j \in \mathcal{V} \setminus \{0\}}$, we perform a sequence of linear combinations to obtain the following $(2N - |\mathcal{V}_{\mathcal{Q}}| - 1)$ unit vectors.

$j \in B$:

$$\begin{aligned} e^{x_j} &= \tfrac{1}{M_j}(v^j - u^j), \quad \text{and} \\ e^{y_j} &= u^j - u^0. \end{aligned}$$

$j \in A$:

If $j \in S_{\mathcal{Q}}$:

$$e^{y_j} = v^j - u^0.$$

If $j \in \overline{S}_{\mathcal{Q}}$:

$$e^{x_j} = \tfrac{1}{\varepsilon}(v^j - u^j).$$

$j \in Z$:

If $j \in \mathcal{N}(\mathcal{Q}, S_{\mathcal{Q}})$:

$$\begin{aligned} e^{y_j} &= v^j - u^j, \quad \text{and} \\ e^{x_j} &= \tfrac{1}{M_j}(u^0 - u^j - e^{y_j} + \textstyle\sum_{i \in B}(M_i e^{x_i} + e^{y_i})). \end{aligned}$$

If $j \in Z \setminus \mathcal{N}(\mathcal{Q}, S_{\mathcal{Q}})$, let $k_j = \operatorname{argmin}\{t(i) : \; i \in \overline{S}_{\mathcal{Q}} \cap \mathcal{P}(j)\}$.

$$\begin{aligned} e^{y_j} &= v^j - u^j, \quad \text{and} \\ e^{x_j} &= \tfrac{1}{M_j}(u^{k_j} + (M_{k_j} - \Delta_{\mathcal{Q}}(k_j))e^{x_{k_j}} - u^j - e^{y_j}). \end{aligned}$$

An additional sequence of linear combinations gives the following additional $|\mathcal{V}_{\mathcal{Q}}|$ vectors.

$\overline{u}^0 = u^0 - \sum_{i \in Z}(M_i e^{x_i} + e^{y_i})$.

$j \in S_{\mathcal{Q}} \setminus \{0\}$,

$$\begin{aligned} \overline{u}^j &= u^j - \textstyle\sum_{i \in Z}(M_i e^{x_i} + e^{y_i}) - \sum_{i \in \mathcal{W}(j)} M_{\mathcal{Q}}(i) e^{x_i} - e^{y_j} \\ &= (\overline{D}_{\mathcal{Q}}(j) - \varepsilon) e^{x_0} + e^{y_0} + \textstyle\sum_{i \in \mathcal{W}(j)} e^{y_i} + \varepsilon e^{x_j}. \end{aligned}$$

$j \in \overline{S}_{\mathcal{Q}}$,

$$\begin{aligned} \overline{v}^j &= v^j - \textstyle\sum_{i \in Z}(M_i e^{x_i} + e^{y_i}) \\ &\qquad - (\Delta_{\mathcal{Q}}(j) + \varepsilon) e^{x_j} - \textstyle\sum_{i \in \mathcal{W}(j)} M_{\mathcal{Q}}(i) e^{x_i} \\ &= (\overline{D}_{\mathcal{Q}}(j) - \Delta_{\mathcal{Q}}(j)) e^{x_0} + e^{y_0} + \textstyle\sum_{i \in \mathcal{W}(j)} e^{y_i} + e^{y_j}. \end{aligned}$$

We now construct a matrix $\mathcal{M}$ whose rows are the $(2N-1)$ vectors $\overline{u}^0$, $\{e^{x_j}\}_{j \in B}$, $\{e^{y_j}\}_{j \in B}$, $\{\overline{u}^j\}_{j \in S_{\mathcal{Q}} \setminus \{0\}}$, $\{e^{y_j}\}_{j \in S_{\mathcal{Q}} \setminus \{0\}}$, $\{e^{x_j}\}_{j \in \overline{S}_{\mathcal{Q}}}$, $\{\overline{v}^j\}_{j \in \overline{S}_{\mathcal{Q}}}$, $\{e^{x_j}\}_{j \in \mathcal{N}(\mathcal{Q}, S_{\mathcal{Q}})}$, $\{e^{y_j}\}_{j \in \mathcal{N}(\mathcal{Q}, S_{\mathcal{Q}})}$, $\{e^{x_j}\}_{j \in Z \setminus \mathcal{N}(\mathcal{Q}, S_{\mathcal{Q}})}$, and $\{e^{y_j}\}_{j \in Z \setminus \mathcal{N}(\mathcal{Q}, S_{\mathcal{Q}})}$. The resulting matrix $\mathcal{M}$ has the following form:

	$\{0\}$		B		$S_{\mathcal{Q}} \setminus \{0\}$		$\overline{S}_{\mathcal{Q}}$		$\mathcal{N}(\mathcal{Q}, S_{\mathcal{Q}})$		$Z \setminus \mathcal{N}(\mathcal{Q}, S_{\mathcal{Q}})$	
	x_0	y_0	x	y	x	y	x	y	x	y	x	y
$\{0\}$	$M_{\mathcal{Q}}(0)$	1										
B			I									
B				I								
$S_{\mathcal{Q}} \setminus \{0\}$	E	1			εI			F				
$S_{\mathcal{Q}} \setminus \{0\}$						I						
$\overline{S}_{\mathcal{Q}}$							I					
$\overline{S}_{\mathcal{Q}}$	G	1						H				
$\mathcal{N}(\mathcal{Q}, S_{\mathcal{Q}})$									I			
$\mathcal{N}(\mathcal{Q}, S_{\mathcal{Q}})$										I		
$Z \setminus \mathcal{N}(\mathcal{Q}, S_{\mathcal{Q}})$											I	
$Z \setminus \mathcal{N}(\mathcal{Q}, S_{\mathcal{Q}})$												I

In the matrix $\mathcal{M}$, the submatrices E and F arise from the nonzero elements of the vectors $\{\overline{u}^j\}_{j \in S_{\mathcal{Q}} \setminus \{0\}}$, and the submatrices G and H arise from the nonzero elements of the vectors $\{\overline{v}^j\}_{j \in \overline{S}_{\mathcal{Q}}}$. Consider the $|\overline{S}_{\mathcal{Q}}| \times |\overline{S}_{\mathcal{Q}}|$ submatrix H. This matrix has a column corresponding to each $j \in \overline{S}_{\mathcal{Q}}$. We arrange the columns of H such that the column corresponding to $i \in \overline{S}_{\mathcal{Q}}$ is before the column corresponding to $j \in \overline{S}_{\mathcal{Q}}$ if $\overline{D}_{\mathcal{Q}}(i) > \overline{D}_{\mathcal{Q}}(j)$ or $t(i) < t(j)$ if

$\overline{D}_{\mathcal{Q}}(i) = \overline{D}_{\mathcal{Q}}(j)$. Note that this arrangement is uniquely defined by assumption (A1) on the set $\mathcal{Q}$. This arrangement guarantees that, for any $j \in \overline{S}_{\mathcal{Q}}$, the column corresponding to $i \in \mathcal{W}(j)$ is before the the column corresponding to j. Consequently, the matrix H is lower-triangular and then it follows that the matrix $\mathcal{M}$ has rank $2N-1$. This is observed by exchanging rows representing $\{\overline{u}^j\}_{j \in S_{\mathcal{Q}} \setminus \{0\}}$ and representing $\{\overline{v}^j\}_{j \in \overline{S}_{\mathcal{Q}}}$, and exchanging columns labelled x in $S_{\mathcal{Q}} \setminus \{0\}$ and y in $\overline{S}_{\mathcal{Q}}$. Since $\mathcal{M}$ was obtained by a sequence of elementary row operations on the $(2N-1) \times 2N$ matrix whose rows are the vectors u^0 and $\{u^j, v^j\}_{j \in \mathcal{V} \setminus \{0\}}$, it follows that these vectors are affinely independent. □

Lemma 4.3 *Consider a feasible solution (x, y) satisfying the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequality at equality. Let $j^* \in \mathcal{F}_{\mathcal{Q}}$ be such that $y_{j^*} = 1$, and let $q \in (\mathcal{Q} \setminus \mathcal{Q}_{q_{j^*}}) \cup \{q_{j^*}\}$, there exists exactly one node $j_q \in \mathcal{F}_{\mathcal{Q}} \cap \mathcal{P}(q)$ such that $y_{j_q} = 1$ and*

(i) $x_i = y_i = 0 \;\; \forall i \in \overline{S}_{\mathcal{Q}} \cap \mathcal{P}(a(j_q))$,

(ii) $x_i = 0 \; \forall i \in \mathcal{P}(a(j_q)) \setminus \mathcal{V}_{\mathcal{Q}_{r_{j_q}}}$ *where* $r_{j_q} = \{i \in \mathcal{Q} : d_{0i} = \widetilde{D}_{\mathcal{Q}}(j_q)\}$,

(iii) $x_i = 0 \; \forall i \in S_{\mathcal{Q}} \cap \mathcal{V}_{\mathcal{Q}}(j_q)$ *and* $y_i = 0 \; \forall i \in \overline{S}_{\mathcal{Q}} \cap \mathcal{V}_{\mathcal{Q}}(j_q)$.

(iv) $\sum_{i \in S_{\mathcal{Q}_{r_{j_q}}}} x_i + \sum_{i \in \overline{S}_{\mathcal{Q}_{r_{j_q}}}} \Delta_{\mathcal{Q}}(i) y_i = \sum_{i \in S_{\mathcal{Q}_{r_{j_q}}}} x_i + \sum_{i \in \overline{S}_{\mathcal{Q}_{r_{j_q}}}} \Delta_{\mathcal{Q}_{r_{j_q}}}(i) y_i = d_{0 r_{j_q}}$.

Proof: For all $q \in \mathcal{Q}$, define $w(q) = \operatorname{argmin}\{t(i) : i \in \bar{S}_{\mathcal{Q}} \cap \mathcal{P}(q) \text{ and } y_i = 1\}$.

First consider $q = Q$. For brevity, let $w = w(Q)$.

Case (a): If w does not exist, then $\sum_{i \in \mathcal{P}(Q)} x_i \geq M_{\mathcal{Q}}(0)$ and $i \notin \overline{S}_{\mathcal{Q}} \; \forall i \in \mathcal{P}(Q)$. Thus, $j^* \notin \mathcal{P}(Q)$ since $j^* \in \overline{S}_{\mathcal{Q}}$ and the left-hand side of the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequality is at least

$$\sum_{i \in \mathcal{P}(Q)} x_i + \overline{D}_{\mathcal{Q}}(j^*) - \widetilde{D}_{\mathcal{Q}}(j^*) > M_{\mathcal{Q}}(0),$$

which contradicts the assumption that the feasible solution satisfies the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequality at equality.

Case (b): If $w \in \mathcal{G}_{\mathcal{Q}}$, then $\sum_{i\in\mathcal{P}(a(w))\cap S_{\mathcal{Q}}} x_i + M_{\mathcal{Q}}(w) \geq d_{0a(w)} + M_{\mathcal{Q}}(w) = M_{\mathcal{Q}}(0)$ since $x_i = y_i = 0\ \forall i \in \mathcal{P}(a(w)) \cap \overline{S}_{\mathcal{Q}}$ by the definition of w. Also, $j^* \neq w$ because $w \in \mathcal{G}_{\mathcal{Q}}$ and $j^* \in \mathcal{F}_{\mathcal{Q}}$. Then the left-hand side of the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequality is at least

$$\sum_{i\in\mathcal{P}(a(w))\cap S_{\mathcal{Q}}} x_i + M_{\mathcal{Q}}(w) + \overline{D}_{\mathcal{Q}}(j^*) - \widetilde{D}_{\mathcal{Q}}(j^*) > M_{\mathcal{Q}}(0),$$

which again gives a contradiction.

Case (c): If $w \in \mathcal{F}_{\mathcal{Q}}$, let $r_w = \{i \in \mathcal{Q} : d_{0i} = \widetilde{D}_{\mathcal{Q}}(w)\}$. Then by Lemmas 4.1 and 4.2, we have

$$\sum_{i\in S_{\mathcal{Q}_{r_w}}} x_i + \sum_{i\in \overline{S}_{\mathcal{Q}_{r_w}}} \Delta_{\mathcal{Q}}(i) y_i \geq \sum_{i\in S_{\mathcal{Q}_{r_w}}} x_i + \sum_{i\in \overline{S}_{\mathcal{Q}_{r_w}}} \Delta_{\mathcal{Q}_{r_w}}(i) y_i \geq d_{0r_w} = \widetilde{D}_{\mathcal{Q}}(w).$$

Thus the left-hand side of the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequality is

$$\begin{aligned} &\geq \sum_{i\in S_{\mathcal{Q}_{r_w}}} x_i + \sum_{i\in \overline{S}_{\mathcal{Q}_{r_w}}} \Delta_{\mathcal{Q}}(i) y_i + \overline{D}_{\mathcal{Q}}(w) - \widetilde{D}_{\mathcal{Q}}(w) && (4.33) \\ &\geq \widetilde{D}_{\mathcal{Q}}(w) + \overline{D}_{\mathcal{Q}}(w) - \widetilde{D}_{\mathcal{Q}}(w) && (4.34) \\ &= \overline{D}_{\mathcal{Q}}(w) = M_{\mathcal{Q}}(0) && (4.35) \end{aligned}$$

Therefore, when the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequality holds at equality, we have the following four properties:

(a) $x_i = y_i = 0\ \ \forall i \in \overline{S}_{\mathcal{Q}} \cap \mathcal{P}(a(w))$,

(b) $x_i = 0\ \forall i \in \mathcal{P}(a(w)) \setminus \mathcal{V}_{\mathcal{Q}_{r_w}}$,

(c) $x_i = 0\ \forall i \in S_{\mathcal{Q}} \cap \mathcal{V}_{\mathcal{Q}}(w)$ and $y_i = 0\ \forall i \in \overline{S}_{\mathcal{Q}} \cap \mathcal{V}_{\mathcal{Q}}(w)$,

(d) $\sum_{i\in S_{\mathcal{Q}_{r_w}}} x_i + \sum_{i\in \overline{S}_{\mathcal{Q}_{r_w}}} \Delta_{\mathcal{Q}}(i) y_i = \sum_{i\in S_{\mathcal{Q}_{r_w}}} x_i + \sum_{i\in \overline{S}_{\mathcal{Q}_{r_w}}} \Delta_{\mathcal{Q}_{r_w}}(i) y_i = d_{0r_w}$,

where (a) follows from the definition of w, (b) and (c) follow from the tightness of the inequality (4.39), and (d) follows from the tightness of the inequality (4.40). Thus, by letting $j_Q = w$, we have proved the claim for $q = Q$.

Now, for any $q \in \{Q-1, \ldots, r_w + 1\}$, we have that $w(q) = w = j_Q$. Thus the claim holds for all such q.

Now consider the case when $q = r_w$. Recall that $\mathcal{Q}_{r_w} = \{1, 2, \dots, r_w\}$. From property (d), we have

$$\sum_{i \in S_{\mathcal{Q}_{r_w}}} x_i + \sum_{i \in \overline{S}_{\mathcal{Q}_{r_w}}} \Delta_{\mathcal{Q}_{r_w}}(i) y_i = d_{0r_w}.$$

Thus the $(\mathcal{Q}_{r_w}, S_{\mathcal{Q}_{r_w}})$ inequality is tight. By proceeding recursively in the above manner, we can show properties (a)-(d) for $\mathcal{Q}_{r_w}$. Note that this recursion terminates when $w = j^*$. Since, otherwise, there must exist a w selected at some step such that $w \in \mathcal{P}(j^*)$, which contradicts property (c) since $y_{j^*} \neq 0$. Since properties (a)-(d) hold at each recursive step and at termination with $w = j^*$, the claim is proven. □

Lemma 4.4 *Consider a feasible solution (x, y) satisfying the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequality at equality. Let $j^* \in \mathcal{G}_{\mathcal{Q}}$ be such that $y_{j^*} = 1$, and let $q \in \mathcal{Q} \setminus \mathcal{Q}_{q_{j^*}}$, there exists exactly one node $j_q \in \mathcal{F}_{\mathcal{Q}} \cap \mathcal{P}(q)$ such that $y_{j_q} = 1$ and*

(i) $x_i = y_i = 0 \quad \forall i \in \overline{S}_{\mathcal{Q}} \cap \mathcal{P}(a(j_q))$,

(ii) $x_i = 0 \; \forall i \in \mathcal{P}(a(j_q)) \setminus \mathcal{V}_{\mathcal{Q}_{r_{j_q}}}$ *where* $r_{j_q} = \{i \in \mathcal{Q} : d_{0i} = \widetilde{D}_{\mathcal{Q}}(j_q)\}$,

(iii) $x_i = 0 \; \forall i \in S_{\mathcal{Q}} \cap \mathcal{V}_{\mathcal{Q}}(j_q)$ *and* $y_i = 0 \; \forall i \in \overline{S}_{\mathcal{Q}} \cap \mathcal{V}_{\mathcal{Q}}(j_q)$.

(iv) $\sum_{i \in S_{\mathcal{Q}_{r_{j_q}}}} x_i + \sum_{i \in \overline{S}_{\mathcal{Q}_{r_{j_q}}}} \Delta_{\mathcal{Q}}(i) y_i = \sum_{i \in S_{\mathcal{Q}_{r_{j_q}}}} x_i + \sum_{i \in \overline{S}_{\mathcal{Q}_{r_{j_q}}}} \Delta_{\mathcal{Q}_{r_{j_q}}}(i) y_i = d_{0r_{j_q}}$.

Proof: For all $q \in \mathcal{Q}$, define $w(q) = \text{argmin}\{t(i) : i \in \bar{S}_{\mathcal{Q}} \cap \mathcal{P}(q) \text{ and } y_i = 1\}$.

First consider $q = Q$. For brevity, let $w = w(Q)$.

Case (a): If w does not exist, then $\sum_{i \in \mathcal{P}(Q)} x_i \geq M_{\mathcal{Q}}(0)$ and $j^* \notin \mathcal{P}(Q)$ since $j^* \in \overline{S}_{\mathcal{Q}}$ and $y_{j^*} = 1$. Then the left-hand side of the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequality is at least

$$\sum_{i \in \mathcal{P}(Q)} x_i + M_{\mathcal{Q}}(j^*) > M_{\mathcal{Q}}(0),$$

which contradicts the assumption that the feasible solution satisfies the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequality at equality.

Case (b): If $w \in \mathcal{G}_{\mathcal{Q}}$, then $\sum_{i \in \mathcal{P}(a(w)) \cap S_{\mathcal{Q}}} x_i + M_{\mathcal{Q}}(w) \geq d_{0a(w)} + M_{\mathcal{Q}}(w) = M_{\mathcal{Q}}(0)$ since $x_i = y_i = 0$ $\forall i \in \mathcal{P}(a(w)) \cap \overline{S}_{\mathcal{Q}}$ by the definition of w. Also, $j^* \neq w$ according to the definition of j^*. Then the left-hand side of the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequality is at least

$$\sum_{i \in \mathcal{P}(a(w)) \cap S_{\mathcal{Q}}} x_i + M_{\mathcal{Q}}(w) + M_{\mathcal{Q}}(j^*) > M_{\mathcal{Q}}(0),$$

which again gives a contradiction.

Case (c): If $w \in \mathcal{F}_{\mathcal{Q}}$, let $r_w = \{i \in \mathcal{Q} : d_{0i} = \widetilde{D}_{\mathcal{Q}}(w)\}$. Then by Lemmas 4.1 and 4.2, we have

$$\sum_{i \in S_{\mathcal{Q}_{r_w}}} x_i + \sum_{i \in \overline{S}_{\mathcal{Q}_{r_w}}} \Delta_{\mathcal{Q}}(i) y_i \geq \sum_{i \in S_{\mathcal{Q}_{r_w}}} x_i + \sum_{i \in \overline{S}_{\mathcal{Q}_{r_w}}} \Delta_{\mathcal{Q}_{r_w}}(i) y_i \geq d_{0r_w} = \widetilde{D}_{\mathcal{Q}}(w).$$

Thus the left-hand side of the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequality is

$$\begin{aligned} &\geq \sum_{i \in S_{\mathcal{Q}_{r_w}}} x_i + \sum_{i \in \overline{S}_{\mathcal{Q}_{r_w}}} \Delta_{\mathcal{Q}}(i) y_i + \overline{D}_{\mathcal{Q}}(w) - \widetilde{D}_{\mathcal{Q}}(w) && (4.36)\\ &\geq \widetilde{D}_{\mathcal{Q}}(w) + \overline{D}_{\mathcal{Q}}(w) - \widetilde{D}_{\mathcal{Q}}(w) && (4.37)\\ &= \overline{D}_{\mathcal{Q}}(w) = M_{\mathcal{Q}}(0) && (4.38) \end{aligned}$$

Therefore, when the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequality holds at equality, we have the following four properties:

(a) $x_i = y_i = 0 \;\; \forall i \in \overline{S}_{\mathcal{Q}} \cap \mathcal{P}(a(w))$,

(b) $x_i = 0 \; \forall i \in \mathcal{P}(a(w)) \setminus \mathcal{V}_{\mathcal{Q}_{r_w}}$,

(c) $x_i = 0 \; \forall i \in S_{\mathcal{Q}} \cap \mathcal{V}_{\mathcal{Q}}(w)$ and $y_i = 0 \; \forall i \in \overline{S}_{\mathcal{Q}} \cap \mathcal{V}_{\mathcal{Q}}(w)$,

(d) $\sum_{i \in S_{\mathcal{Q}_{r_w}}} x_i + \sum_{i \in \overline{S}_{\mathcal{Q}_{r_w}}} \Delta_{\mathcal{Q}}(i) y_i = \sum_{i \in S_{\mathcal{Q}_{r_w}}} x_i + \sum_{i \in \overline{S}_{\mathcal{Q}_{r_w}}} \Delta_{\mathcal{Q}_{r_w}}(i) y_i = d_{0r_w}$,

where (a) follows from the definition of w, (b) and (c) follow from the tightness of the inequality (4.39), and (d) follows from the tightness of the inequality (4.40). Thus, by letting $j_Q = w$, we have proved the claim for $q = Q$.

Now, for any $q \in \{Q-1, \ldots, r_w + 1\}$, we have that $w(q) = w = j_Q$. Thus the claim holds for all such q.

Now consider the case when $q = r_w$. Recall that $\mathcal{Q}_{r_w} = \{1, 2, \ldots, r_w\}$. From property (d), we have

$$\sum_{i \in S_{\mathcal{Q}_{r_w}}} x_i + \sum_{i \in \overline{S}_{\mathcal{Q}_{r_w}}} \Delta_{\mathcal{Q}_{r_w}}(i) y_i = d_{0r_w}.$$

Thus the $(\mathcal{Q}_{r_w}, S_{\mathcal{Q}_{r_w}})$ inequality is tight. By proceeding recursively in the above manner, we can show properties (a)-(d) for $\mathcal{Q}_{r_w}$. Note that this recursion terminates when $q = q_{j^*}$. Since, otherwise, there must exist a w selected at some step such that $w \in \mathcal{P}(j^*)$, which contradicts property (c) since $y_{j^*} \neq 0$. Since properties (a)-(d) hold at each recursive step and at termination with $q = q_{j^*}$, the claim is proven. $\square$

Lemma 4.5 *Consider a feasible solution (x, y) satisfying the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequality at equality. Let $j^* \in S_{\mathcal{Q}}$ be such that $x_{j^*} > 0$, and let $q \in \mathcal{Q} \setminus \mathcal{Q}_{q_{j^*}}$, there exists exactly one node $j_q \in \mathcal{F}_{\mathcal{Q}} \cap \mathcal{P}(q)$ such that $y_{j_q} = 1$ and*

(i) $x_i = y_i = 0 \ \ \forall i \in \overline{S}_{\mathcal{Q}} \cap \mathcal{P}(a(j_q))$,

(ii) $x_i = 0 \ \forall i \in \mathcal{P}(a(j_q)) \setminus \mathcal{V}_{\mathcal{Q}_{r_{j_q}}}$ *where* $r_{j_q} = \{i \in \mathcal{Q} : d_{0i} = \widetilde{D}_{\mathcal{Q}}(j_q)\}$,

(iii) $x_i = 0 \ \forall i \in S_{\mathcal{Q}} \cap \mathcal{V}_{\mathcal{Q}}(j_q)$ *and* $y_i = 0 \ \forall i \in \overline{S}_{\mathcal{Q}} \cap \mathcal{V}_{\mathcal{Q}}(j_q)$.

(iv) $\sum_{i \in S_{\mathcal{Q}_{r_{j_q}}}} x_i + \sum_{i \in \overline{S}_{\mathcal{Q}_{r_{j_q}}}} \Delta_{\mathcal{Q}}(i) y_i = \sum_{i \in S_{\mathcal{Q}_{r_{j_q}}}} x_i + \sum_{i \in \overline{S}_{\mathcal{Q}_{r_{j_q}}}} \Delta_{\mathcal{Q}_{r_{j_q}}}(i) y_i = d_{0r_{j_q}}$.

Proof: For all $q \in \mathcal{Q}$, define $w(q) = \text{argmin}\{t(i) : i \in \bar{S}_{\mathcal{Q}} \cap \mathcal{P}(q) \text{ and } y_i = 1\}$.

First consider $q = Q$. For brevity, let $w = w(Q)$.

Case (a): If w does not exist, then $\sum_{i \in \mathcal{P}(Q)} x_i \geq M_{\mathcal{Q}}(0)$ and $j^* \notin \mathcal{P}(Q)$ according to the definition of j^*. Then the left-hand side of the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequality is at least

$$\sum_{i \in \mathcal{P}(Q)} x_i + x_{j^*} > M_{\mathcal{Q}}(0),$$

which contradicts the assumption that the feasible solution satisfies the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequality at equality.

Case (b): If $w \in \mathcal{G}_{\mathcal{Q}}$, then $\sum_{i \in \mathcal{P}(a(w)) \cap S_{\mathcal{Q}}} x_i + M_{\mathcal{Q}}(w) \geq d_{0a(w)} + M_{\mathcal{Q}}(w) = M_{\mathcal{Q}}(0)$ since $x_i = y_i = 0 \; \forall i \in \mathcal{P}(a(w)) \cap \overline{S}_{\mathcal{Q}}$ by the definition of w. Also, $j^* \neq w$ according to the definition of j^*. Then the left-hand side of the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequality is at least

$$\sum_{i \in \mathcal{P}(a(w)) \cap S_{\mathcal{Q}}} x_i + M_{\mathcal{Q}}(w) + x_{j^*} > M_{\mathcal{Q}}(0),$$

which again gives a contradiction.

Case (c): If $w \in \mathcal{F}_{\mathcal{Q}}$, let $r_w = \{i \in \mathcal{Q} : d_{0i} = \widetilde{D}_{\mathcal{Q}}(w)\}$. Then by Lemmas 4.1 and 4.2, we have

$$\sum_{i \in S_{\mathcal{Q}_{r_w}}} x_i + \sum_{i \in \overline{S}_{\mathcal{Q}_{r_w}}} \Delta_{\mathcal{Q}}(i) y_i \geq \sum_{i \in S_{\mathcal{Q}_{r_w}}} x_i + \sum_{i \in \overline{S}_{\mathcal{Q}_{r_w}}} \Delta_{\mathcal{Q}_{r_w}}(i) y_i \geq d_{0r_w} = \widetilde{D}_{\mathcal{Q}}(w).$$

Thus the left-hand side of the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequality is

$$\begin{aligned} &\geq \sum_{i \in S_{\mathcal{Q}_{r_w}}} x_i + \sum_{i \in \overline{S}_{\mathcal{Q}_{r_w}}} \Delta_{\mathcal{Q}}(i) y_i + \overline{D}_{\mathcal{Q}}(w) - \widetilde{D}_{\mathcal{Q}}(w) && (4.39) \\ &\geq \widetilde{D}_{\mathcal{Q}}(w) + \overline{D}_{\mathcal{Q}}(w) - \widetilde{D}_{\mathcal{Q}}(w) && (4.40) \\ &= \overline{D}_{\mathcal{Q}}(w) = M_{\mathcal{Q}}(0) && (4.41) \end{aligned}$$

Therefore, when the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequality holds at equality, we have the following four properties:

(a) $x_i = y_i = 0 \;\; \forall i \in \overline{S}_{\mathcal{Q}} \cap \mathcal{P}(a(w))$,

(b) $x_i = 0 \; \forall i \in \mathcal{P}(a(w)) \setminus \mathcal{V}_{\mathcal{Q}_{r_w}}$,

(c) $x_i = 0 \; \forall i \in S_{\mathcal{Q}} \cap \mathcal{V}_{\mathcal{Q}}(w)$ and $y_i = 0 \; \forall i \in \overline{S}_{\mathcal{Q}} \cap \mathcal{V}_{\mathcal{Q}}(w)$,

(d) $\sum_{i \in S_{\mathcal{Q}_{r_w}}} x_i + \sum_{i \in \overline{S}_{\mathcal{Q}_{r_w}}} \Delta_{\mathcal{Q}}(i) y_i = \sum_{i \in S_{\mathcal{Q}_{r_w}}} x_i + \sum_{i \in \overline{S}_{\mathcal{Q}_{r_w}}} \Delta_{\mathcal{Q}_{r_w}}(i) y_i = d_{0r_w}$,

where (a) follows from the definition of w, (b) and (c) follow from the tightness of the inequality (4.39), and (d) follows from the tightness of the inequality (4.40). Thus, by letting $j_Q = w$, we have proved the claim for $q = Q$.

Now, for any $q \in \{Q - 1, \ldots, r_w + 1\}$, we have that $w(q) = w = j_Q$. Thus the claim holds for all such q.

Now consider the case when $q = r_w$. Recall that $\mathcal{Q}_{r_w} = \{1, 2, \ldots, r_w\}$. From property (d), we have

$$\sum_{i \in S_{\mathcal{Q}_{r_w}}} x_i + \sum_{i \in \overline{S}_{\mathcal{Q}_{r_w}}} \Delta_{\mathcal{Q}_{r_w}}(i) y_i = d_{0r_w}.$$

Thus the $(\mathcal{Q}_{r_w}, S_{\mathcal{Q}_{r_w}})$ inequality is tight. By proceeding recursively in the above manner, we can show properties (a)-(d) for $\mathcal{Q}_{r_w}$. Note that this recursion can be terminated when find a $w \in \mathcal{V}(j^*)$ or $q = q_{j^*}$. Since, otherwise, there must exist a w selected at some step such that $w \in \mathcal{P}(j^*)$, which contradicts property (c) since $x_{j^*} > 0$. Since properties (a)-(d) hold at each recursive step and at termination with $q = q_{j^*}$, the claim is proven. □

Proof of necessity.

We consider in turn the conditions (i)-(iv) and show that if any condition is removed, the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequality is not facet-defining.

Condition (i): The proof is by contradiction. Suppose $0 \in \overline{S}_{\mathcal{Q}}$. Since $y_0 = 1$ and $\Delta_{\mathcal{Q}}(0) = M_{\mathcal{Q}}(0)$, then we have $x_i = 0\ \forall i \in S_{\mathcal{Q}} \setminus \{0\}$ and $y_i = 0\ \forall i \in \overline{S}_{\mathcal{Q}}$ in order to satisfy the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequality at equality. Thus, $\dim(X_{SLSF}) \leq 2\mathcal{N} - 2 - |S_{\mathcal{Q}} \setminus \{0\}| - |\overline{S}_{\mathcal{Q}}| < 2\mathcal{N} - 2$, where X_{SLSF} is the set of feasible solutions satisfying the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequality at equality.

Condition (ii): The proof is by contradiction. Suppose there is a node $j^* \in \mathcal{N}(\mathcal{Q}, S_{\mathcal{Q}})$ such that $M_{\mathcal{Q}}(0) < d_{0j^*}$. Let $w = \{i \in \mathcal{V}_{\mathcal{Q}} : j^* \in \mathcal{C}(i)\}$. Then

$$M_{\mathcal{Q}}(0) = \sum_{i \in S_{\mathcal{Q}}} x_i + \sum_{i \in \overline{S}_{\mathcal{Q}}} \Delta_{\mathcal{Q}}(i) y_i \geq \sum_{i \in \mathcal{P}(w)} x_i$$

since $i \in S_{\mathcal{Q}}$ for all $i \in \mathcal{P}(w)$ by the definition of $\mathcal{N}(\mathcal{Q}, S_{\mathcal{Q}})$. Then, $\sum_{i \in \mathcal{P}(w)} x_i \leq M_{\mathcal{Q}}(0) < d_{0j^*}$. Thus, we have $y_{j^*} = 1$ for all feasible solutions satisfying the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequality at equality and $\dim(X_{SLSF}) < 2\mathcal{N} - 2$.

Condition (iii): The proof of (a) is by contradiction. Suppose $q^* = \text{argmax}\{i \in \mathcal{Q} :$

$\mathcal{W}(j) \cap \mathcal{P}(i) = \emptyset\}$. Then we have

$$\sum_{i \in S_{\mathcal{Q} \setminus \mathcal{Q}_{q^*-1}}} x_i + \sum_{i \in \overline{S}_{\mathcal{Q} \setminus \mathcal{Q}_{q^*-1}}} \Delta_{\mathcal{Q} \setminus \mathcal{Q}_{q^*-1}}(i) y_i \geq M_{\mathcal{Q}}(0)$$

corresponding to leaf node set $\mathcal{Q} \setminus \mathcal{Q}_{q^*-1}$ since $i \in S_{\mathcal{Q}}$ for all $i \in \mathcal{P}(q^*)$. Thus, $x_i = 0$ for all $i \in S_{\mathcal{Q}_{q^*}} \setminus \mathcal{P}(q^*)$ and $y_i = 0$ for all $i \in \overline{S}_{\mathcal{Q}_{q^*}} \setminus \mathcal{P}(q^*)$ are required for the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequality to be tight. This implies $\dim(X_{SLSF}) < 2\mathcal{N} - 2$.

The proof of (b) is by contradiction. Suppose $y_{j^*} = 1$ for some feasible solution satisfying the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequality at equality, we will prove that $\widetilde{D}_{\mathcal{Q}}(j^*) \geq d_{0a(k)}$ for all $k \in \mathcal{W}(j^*)$, which implies that if $\exists k \in \mathcal{W}(j^*)$ such that $\widetilde{D}_{\mathcal{Q}}(j^*) < d_{0a(k)}$, then $y_{j^*} = 0$ for any feasible solution satisfying the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequality at equality and $\dim(X_{SLSF}) < 2N - 2$.

Now suppose $y_{j^*} = 1$ for some feasible solution satisfying the inequality at equality. Let $G(j^*)$ be the set of nodes $\{j_Q, j_{r_w}, \ldots\}$ identified at each recursive step in the proof of Lemma 4.3 from the leaf nodes $\{Q, r_w, \ldots\}$ except for the termination step. Define $u_j = \text{argmax}\{t(i) : i \in \mathcal{P}(j) \cap \mathcal{P}(j^*)\}\ \forall j \in G(j^*)$ and $u_{j^*} = \text{argmax}\{t(i) : i \in \mathcal{P}(r_{j^*}) \cap \mathcal{P}(j^*)\}$. From property (iv) in Lemma 4.3,

$$\begin{aligned}
\widetilde{D}_{\mathcal{Q}}(j^*) &= \sum_{i \in S_{\mathcal{Q}_{r_{j^*}}}} x_i + \sum_{i \in \overline{S}_{\mathcal{Q}_{r_{j^*}}}} \Delta_{\mathcal{Q}_{r_{j^*}}}(i) y_i && (4.42)\\
&\geq \sum_{i \in S_{\mathcal{Q}_{r_{j^*}}} \cap \mathcal{P}(u_{j^*})} x_i + \sum_{i \in \overline{S}_{\mathcal{Q}_{r_{j^*}}} \cap \mathcal{P}(u_{j^*})} \Delta_{\mathcal{Q}_{r_{j^*}}}(i) y_i && (4.43)\\
&= \sum_{i \in \mathcal{P}(u_{j^*})} x_i && (4.44)\\
&= \sum_{i \in \mathcal{P}(a(j^*))} x_i, && (4.45)
\end{aligned}$$

where (4.44) follows from property (i) of Lemma 4.3 and (4.45) follows from property (ii) of Lemma 4.3 as $j_q = j^*$. Thus

$$\widetilde{D}_{\mathcal{Q}}(j^*) \geq \sum_{i \in \mathcal{P}(a(j^*))} x_i \geq \sum_{i \in \mathcal{P}(u_j)} x_i \geq \sum_{i \in \mathcal{P}(a(j))} x_i \geq d_{0a(j)}\ \forall j \in G(j^*), \qquad (4.46)$$

where the third inequality follows from property (ii) of Lemma 4.3.

Finally, from the definition of $\mathcal{W}(j^*)$, we have $\mathcal{W}(j^*) \cap \mathcal{P}(q) \in \mathcal{P}(G(j^*) \cap \mathcal{P}(q))$ $\forall q \in \mathcal{Q} \setminus \mathcal{Q}_{q_{j^*}}$. Then, $\widetilde{D}_{\mathcal{Q}}(j^*) \geq d_{0a(k)}$ $\forall k \in \mathcal{W}(j^*)$.

Similar as the proof of (b), the proof of (c) is also by contradiction. Suppose $y_{j^*} = 1$ for some feasible solution satisfying the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequality at equality, we only need to prove that $d_{0a(j^*)} \geq d_{0a(k)}$ for all $k \in \mathcal{W}(j^*)$.

Now suppose $y_{j^*} = 1$ for some feasible solution satisfying the inequality at equality. Let $G(j^*)$ be the set of nodes $\{j_Q, j_{r_w}, \ldots\}$ identified at each recursive step in the proof of Lemma 4.4 from the leaf nodes $\{Q, r_w, \ldots\}$. Define $u_j = \text{argmax}\{t(i) : i \in \mathcal{P}(j) \cap \mathcal{P}(j^*)\}$ $\forall j \in G(j^*)$. From property (iv) in Lemma 4.4, for each $j \in G(j^*)$, we have

$$\overline{D}_{\mathcal{Q}}(j^*) = \sum_{i \in S_{\mathcal{Q}_{q_{j^*}}}} x_i + \sum_{i \in \overline{S}_{\mathcal{Q}_{q_{j^*}}}} \Delta_{\mathcal{Q}_{q_{j^*}}}(i) y_i \tag{4.47}$$

$$\geq \sum_{i \in S_{\mathcal{Q}_{q_{j^*}}} \cap \mathcal{P}(u_j)} x_i + \sum_{i \in \overline{S}_{\mathcal{Q}_{q_{j^*}}} \cap \mathcal{P}(u_j)} \Delta_{\mathcal{Q}_{q_{j^*}}}(i) y_i + M_{\mathcal{Q}}(j^*) \tag{4.48}$$

$$\geq \sum_{i \in \mathcal{P}(u_j)} x_i + M_{\mathcal{Q}}(j^*). \tag{4.49}$$

where (4.49) follows from property (i) of Lemma 4.4. Then, $\sum_{i \in \mathcal{P}(u_j)} x_i \leq d_{0a(j^*)}$ based on (4.49). According to property (ii) of Lemma 4.4, we have

$$d_{0a(j^*)} \geq \sum_{i \in \mathcal{P}(u_j)} x_i = \sum_{i \in \mathcal{P}(a(j))} x_i \geq d_{0a(j)}.$$

Finally, from the definition of $\mathcal{W}(j^*)$, we have $\mathcal{W}(j^*) \cap \mathcal{P}(q) \in \mathcal{P}(G(j^*) \cap \mathcal{P}(q))$ $\forall q \in \mathcal{Q} \setminus \mathcal{Q}_{q_{j^*}}$. Then, $d_{0a(j^*)} \geq d_{0a(k)}$ $\forall k \in \mathcal{W}(j^*)$.

Similar as the proof of (b), the proof of (d) is also by contradiction. Suppose $x_{j^*} > 0$ for some feasible solution satisfying the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequality at equality, we only need to prove that $\overline{D}_{\mathcal{Q}}(j^*) \geq d_{0a(k)}$ for all $k \in \mathcal{W}(j^*)$.

Now suppose $x_{j^*} > 0$ for some feasible solution satisfying the inequality at equality. Let $G(j^*)$ be the set of nodes $\{j_Q, j_{r_w}, \ldots\}$ identified at each recursive step in the proof of Lemma 4.5 from the leaf nodes $\{Q, r_w, \ldots\}$. Define $u_j = \text{argmax}\{t(i) : i \in \mathcal{P}(j) \cap$

$\mathcal{P}(j^*)\}\ \forall j \in G(j^*)$. From property (iv) in Lemma 4.5, for each $j \in G(j^*)$, we have

$$\begin{aligned} \overline{D}_{\mathcal{Q}}(j^*) &= \sum_{i \in S_{\mathcal{Q}_{q_{j^*}}}} x_i + \sum_{i \in \overline{S}_{\mathcal{Q}_{q_{j^*}}}} \Delta_{\mathcal{Q}_{q_{j^*}}}(i) y_i & (4.50) \\ &\geq \sum_{i \in S_{\mathcal{Q}_{q_{j^*}}} \cap \mathcal{P}(u_j)} x_i + \sum_{i \in \overline{S}_{\mathcal{Q}_{q_{j^*}}} \cap \mathcal{P}(u_j)} \Delta_{\mathcal{Q}_{q_{j^*}}}(i) y_i + x_{j^*} & (4.51) \\ &\geq \sum_{i \in \mathcal{P}(u_j)} x_i + x_{j^*}. & (4.52) \end{aligned}$$

where (4.52) follows from property (i) of Lemma 4.5. Then, $\sum_{i \in \mathcal{P}(u_j)} x_i < \overline{D}_{\mathcal{Q}}(j^*)$ based on (4.52). According to property (ii) of Lemma 4.5, we have

$$\overline{D}_{\mathcal{Q}}(j^*) > \sum_{i \in \mathcal{P}(u_j)} x_i = \sum_{i \in \mathcal{P}(a(j))} x_i \geq d_{0a(j)}.$$

Finally, from the definition of $\mathcal{W}(j^*)$, we have $\mathcal{W}(j^*) \cap \mathcal{P}(q) \in \mathcal{P}(G(j^*) \cap \mathcal{P}(q))\ \forall q \in \mathcal{Q} \setminus \mathcal{Q}_{q_{j^*}}$. Then, $\overline{D}_{\mathcal{Q}}(j^*) > d_{0a(k)}\ \forall k \in \mathcal{W}(j^*)$.

Condition (iv): The proof is by contradiction. Suppose, for some $j \in \mathcal{G}_{\mathcal{Q}}$, there exists a $\overline{q} \in \mathcal{L} \cap \mathcal{Q}$ such that $\overline{q} = \text{argmax}\{q : q \in \mathcal{Q}(j)\}$. Now consider the values of x_j and y_j for any feasible solution satisfying the inequality at equality. If $y_j = 0$, then $x_j = 0$. If $y_j = 1$, then from the recursion in the proof of (c) in condition (iii), we have $\sum_{i \in \mathcal{P}(a(j))} x_i = M_{\mathcal{Q}_{\overline{q}}}(0) - M_{\mathcal{Q}_{\overline{q}}}(j)$, which implies that $x_j \geq M_{\mathcal{Q}_{\overline{q}}}(j) = M_{\mathcal{Q}}(j) = M_j$ in order to keep feasibility since $x_i = 0\ \forall i \in \mathcal{V}_{\mathcal{Q}}(j)$, which implies $x_j = M_j$. Thus, we have $x_j = M_j y_j$, which is independent of $y_0 = 1$ and $\sum_{i \in S_{\mathcal{Q}}} x_i + \sum_{i \in \overline{S}_{\mathcal{Q}}} \Delta_{\mathcal{Q}}(i) y_i = M_{\mathcal{Q}}(0)$ so that $\dim(X_{SLSF}) < 2N - 2$. □

Example (continued): Consider the three inequalities added in the example. The first one is not facet-defining since $0 \notin S_{\mathcal{Q}}$. The second one is not facet-defining since it does not satisfy condition (ii) of Theorem 4.2. However, the third inequality is facet-defining. To illustrate the necessity of condition (iii), the inequality $x_0 + x_1 + x_4 + x_3 + 10y_6 \geq 45$, where $\mathcal{Q} = \{4, 6\}$ and $\overline{S}_{\mathcal{Q}} = \{6\}$, is not facet-defining since $\overline{D}_{\mathcal{Q}}(4) = d_{0a(6)}$ and $6 \in \mathcal{W}(4)$, which contradicts condition (d) of (iii). On the other hand, the inequality $x_0 + x_1 + x_4 + x_2 + 10y_5 \geq$

45, where $\mathcal{Q} = \{4, 5\}$ and $\overline{S}_{\mathcal{Q}} = \{5\}$, satisfies all four conditions of Theorem 4.2 and therefore is facet-defining.

Recall that every (ℓ, S) inequality is a $(\mathcal{Q}, S_{\mathcal{Q}})$ inequality with $\mathcal{Q} = \{\ell\}$ and $S_{\mathcal{Q}} = S$. We then have the following corollary to Theorem 4.2.

Corollary 4.1 *An (ℓ, S) inequality is facet-defining if and only if ℓ and S are such that*

(i) $0 \in S_{\mathcal{Q}}$,

(ii) $d_{0\ell} \geq \max_{i \in N(\ell,S)} d_{0i}$,

(iv) $\mathcal{P}(\ell) \setminus S \neq \emptyset$, $\ell \notin \mathcal{L}$ *or* $\mathcal{P}(\ell) \setminus S = \emptyset$, $\ell \in \mathcal{L}$.

In this case, the neighborhood is simply $N(\ell, S) = \{j : j \in \mathcal{C}(i) \setminus \mathcal{P}(\ell)$ where $i < \operatorname{argmin}\{t(k) : k \in \overline{S}\}\}$, and condition (iii) is redundant since $\mathcal{W}(j) = \emptyset$ for all $j \in \mathcal{V}_{\mathcal{Q}}$.

Remark 4.1 *From above, we can see that $(\mathcal{Q}, S_{\mathcal{Q}})$ inequalities suffice to describe the convex hull of the deterministic case of (SLS) since in this case, they are equivalent to the (ℓ, S) inequalities. Moreover, the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequalities are also sufficient to describe the convex hull when (SLS) has two periods as shown in Chapter 3, which lies in the fact that the inequalities (3.22) developed in Chapter 3 is a special case of $(\mathcal{Q}, S_{\mathcal{Q}})$ inequalities with $S_{\mathcal{Q}} = \{0, i_1\}$ and $\overline{S}_{\mathcal{Q}} = \mathcal{Q} \setminus S_{\mathcal{Q}}$ for the given set $\mathcal{Q} \subseteq \mathcal{C}(0)$.*

4.4 Separation of $(\mathcal{Q}, S_{\mathcal{Q}})$ Inequalities

Given the set $\mathcal{Q}$, and a fractional solution (x^*, y^*) of (SLS), let

$$S^*_{\mathcal{Q}} = \{i \in \mathcal{V}_{\mathcal{Q}} : \; x^*_i \leq \Delta_{\mathcal{Q}}(i) y^*_i\}. \tag{4.53}$$

If $\sum_{i \in S^*_{\mathcal{Q}}} x^*_i + \sum_{i \in \overline{S}^*_{\mathcal{Q}}} \Delta_{\mathcal{Q}}(i) y^*_i < M_{\mathcal{Q}}(0)$, then the $(\mathcal{Q}, S^*_{\mathcal{Q}})$ inequality is violated. On the other hand, if (x^*, y^*) satisfies the $(\mathcal{Q}, S^*_{\mathcal{Q}})$ inequality then there are no violated $(\mathcal{Q}, S_{\mathcal{Q}})$ inequalities corresponding to the node set $\mathcal{Q}$, since

$$\min_{S_{\mathcal{Q}} \subseteq \mathcal{V}_{\mathcal{Q}}} \left\{ \sum_{i \in S_{\mathcal{Q}}} x^*_i + \sum_{i \in \overline{S}_{\mathcal{Q}}} \Delta_{\mathcal{Q}}(i) y^*_i \right\} = \sum_{i \in S^*_{\mathcal{Q}}} x^*_i + \sum_{i \in \overline{S}^*_{\mathcal{Q}}} \Delta_{\mathcal{Q}}(i) y^*_i \geq M_{\mathcal{Q}}(0).$$

The difficulty in separating $(\mathcal{Q}, S_{\mathcal{Q}})$ inequalities is how to determine $\mathcal{Q}$. The $(\mathcal{Q}, S_{\mathcal{Q}})$ inequalities with $|\mathcal{Q}| = Q$ can be separated in $\mathcal{O}(N^{Q+1})$ time by enumeration since for each such $\mathcal{Q}$, we can check for a violated $(\mathcal{Q}, S_{\mathcal{Q}})$ inequality in $\mathcal{O}(N)$ time. Because we don't know a polynomial algorithm for general $\mathcal{Q}$, we only check all of the $|\mathcal{Q}| = 1$ and $|\mathcal{Q}| = 2$ inequalities for violations and then we apply a heuristic (Algorithm 2) to try to find some violated inequalities for larger Q.

The basic idea of Algorithm 2 is to add nodes to $\mathcal{Q}$, using a depth-first strategy, such that the right-hand-side of the inequality is not changed while the left-hand-side decreases. The process stops as soon as we find a violated $(\mathcal{Q}, S^*_{\mathcal{Q}})$ inequality. If no violated inequality is found after exhausting the depth-first search, we re-start the search with a new node.

Algorithm 2 Heuristic separation of $\{\mathcal{Q}, S_{\mathcal{Q}}\}$ inequalities with $|\mathcal{Q}| \geq 3$

Input: a fractional solution (x^*, y^*).
for $\ell \in \mathcal{V}$ **do**
  Step 0. Set $\mathcal{Q} = \{\ell\}$ and $i = \ell$.
  Step 1. If $|\mathcal{Q}| \geq 3$, go to Step 2. Otherwise, go to Step 3.
  Step 2. Compute $S^*_{\mathcal{Q}}$ as in (4.53). If the $(\mathcal{Q}, S^*_{\mathcal{Q}})$ inequality is violated **stop**.
  Step 3. For some node $j \in \mathcal{V}(a(i)) \setminus \mathcal{V}(i)$, let $\mathcal{Q}' = \mathcal{Q} \cup \{j\}$. If a node $k = \text{argmax}\{d_{0j} : \; j \in \mathcal{V}(a(i)) \setminus \mathcal{V}(i),\; d_{0j} < d_{0i}$ and $\sum_{i \in S^*_{\mathcal{Q}'}} x^*_i + \sum_{i \in \overline{S}^*_{\mathcal{Q}'}} \Delta_{\mathcal{Q}'}(i) y^*_i < \sum_{i \in S^*_{\mathcal{Q}}} x^*_i + \sum_{i \in \overline{S}^*_{\mathcal{Q}}} \Delta_{\mathcal{Q}}(i) y^*_i\}$ exists, go to Step 5. Otherwise, go to Step 4.
  Step 4. If $i \neq 0$, set $i \leftarrow a(i)$ and **go to** Step 3. If $i = 0$ **end for**.
  Step 5. Set $\mathcal{Q} \leftarrow \mathcal{Q} \cup \{k\}$ and $i \leftarrow k$ and **go to** Step 1.
end for

4.5 Computational Experiments

In this section, we report on the computational effectiveness of the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequalities on randomly generated instances of single-item, uncapacitated, stochastic lot-sizing problems.

4.5.1 Implementation

We implemented a branch-and-cut scheme in which complete separation of $(\mathcal{Q}, S_{\mathcal{Q}})$ inequalities is done for $|\mathcal{Q}| = 1$ and $|\mathcal{Q}| = 2$ followed by Algorithm 2. We add all violated $|\mathcal{Q}| = 1$ inequalities if some are found and repeat until no more are found. We do the same for $|\mathcal{Q}| = 2$ inequalities. When no more of these are found, we apply Algorithm 2 and add inequalities one-at-a-time until no further violation is found.

Our implementation was carried out in C using the callable libraries of CPLEX 8.1. Default CPLEX options were used throughout. All computations were carried out on a 2.4GHz Intel Xeon/Linux workstation with 2GB RAM with one hour time limit per run.

4.5.2 Test Problem Generation

A number of instances of (SLS) were generated corresponding to different structures of the underlying scenario trees, different ratios of the production cost to the inventory holding cost, and different ratios of the setup cost to the inventory holding cost.

We assumed that the underlying scenario tree is balanced with T stages and K branches per stage. We considered 6 different tree structures with $K = 2$ and $T \in \{10, 11\}$; $K = 3$ and $T \in \{6, 7\}$; $K = 4$ and $T \in \{5, 6\}$. We considered three different levels of production to holding cost ratio $\alpha/h \in \{50, 100, 200\}$, and three different levels of setup to holding cost ratio $\beta/h \in \{1750, 3500, 7000\}$.

For each of the 54 combinations of the tree structure, α/h and β/h, we generated three random instances as follows. For each node i of the tree, the holding cost h_i is sampled from $U[0.01, 0.05]$, i.e., a uniform random number in the interval $[0.01, 0.05]$; α_i is sampled from $U[0.8(\alpha/h)\bar{h}, 1.2(\alpha/h)\bar{h}]$ where $\bar{h} = 0.03$ is the average holding cost; $\beta_i \sim U[0.8(\beta/h)\bar{h}, 1.2(\beta/h)\bar{h}]$; and $d_i \sim U[10, 100]$. Finally, all K children of a node were assigned equal probabilities.

4.5.3 Results

Tables 2, 3, and 4 report on the effectiveness of the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequalities in tightening the LP relaxation gap for the instances corresponding to $K = 2, 3$ and 4 at the root node. The column labelled LP Gap % gives the relative LP relaxation gap of the original formulation (SLS) with respect to the best feasible solution found with our branch-and-cut scheme. The columns labelled $|\mathcal{Q}| = 1$, $|\mathcal{Q}| = 2$ and General $\mathcal{Q}$ correspond to the results from adding all violated $(\mathcal{Q}, S_{\mathcal{Q}})$ inequalities for $|\mathcal{Q}| = 1$ and then those for $|\mathcal{Q}| = 2$, and then heuristically for some violated inequalities with $|\mathcal{Q}| > 2$. For each combination of T, β/h and α/h, there are two rows corresponding to the columns labelled $|\mathcal{Q}| = 1$, $|\mathcal{Q}| = 2$ and General $\mathcal{Q}$. The first row gives the LP relaxation gap after adding the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequalities, and the second row gives the number of $(\mathcal{Q}, S_{\mathcal{Q}})$ inequalities added. Note that all reported numbers are averages over three instances. Significant tightening of the LP relaxation is achieved via the proposed $(\mathcal{Q}, S_{\mathcal{Q}})$ inequalities. In some cases, the LP relaxation gap is reduced from over 20% to 0.4%. Furthermore, in most cases, the LP relaxation gap is small after adding the inequalities corresponding to $|\mathcal{Q}| = 1$ and $|\mathcal{Q}| = 2$.

The results from our branch-and-cut scheme are reported in Tables 5, 6, and 7 for the instances corresponding to $K = 2, 3$ and 4, respectively. For each combination of T, β/h and α/h, there are two rows. The first row gives the performance of the default CPLEX MIP solver and the second row gives the performance of our branch-and-cut scheme. We give the number of cutting planes added by the default CPLEX MIP solver and by our branch-and-cut scheme respectively, the relative optimality gap upon termination, the number of nodes explored (apart from the root node), and the total CPU time. The reported data is averaged over three instances. The numbers in square brackets indicate the number of instances *not* solved to default CPLEX optimality tolerance within the allotted time limit of one hour. The default CPLEX MIP solver adds several types of cuts including flow covers, Gomory fractional cuts and mixed integer rounding cuts. Our branch-and-cut algorithm adds $(\mathcal{Q}, S_{\mathcal{Q}})$ cuts at each node after the CPLEX default cuts have been added. For the total CPU time, we report the average CPU time for instances that are solved to default CPLEX optimality tolerance within the allotted time limit of one hour. Otherwise, we use "$_{***}$" to represent the case that *no* instance is solved to default CPLEX optimality tolerance within the allotted time. The efficiency of the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequalities within our branch-and-cut is clearly observed. Our branch-and-cut algorithm proves optimality for all instances for $K = 2$, has only 11 and 25 instances unsolved to optimality for $K = 3$ and $K = 4$, respectively. In contrast, the unsolved instances corresponding to default CPLEX are 6, 43 and 52, respectively. For cases where neither algorithm could prove optimality, our algorithm yielded much smaller optimality gaps. Moreover, our cuts dramatically reduced the number of nodes in the tree and, although we added more cuts, the running times were smaller as well. Because we add so many $(\mathcal{Q}, S_{\mathcal{Q}})$ inequalities, we thought that the running times might be reduced substantially by deleting cuts that were no longer tight. However, experiments using cut management did not yield significant improvement.

Table 2: Results for the root node ($K = 2$)

T	β/h	α/h	LP Gap %	$\|\mathcal{Q}\| = 1$	$\|\mathcal{Q}\| = 2$	General $\mathcal{Q}$
10	1750	50	7.19	0.04	0.01	0.01
				3473	1185	18
10	1750	100	6.60	0.04	0.00	0.00
				3492	1238	19
10	1750	200	5.28	0.04	0.00	0.00
				3451	1124	0
10	3500	50	13.06	0.11	0.01	0.01
				3424	2513	51
10	3500	100	12.10	0.10	0.01	0.01
				3374	2630	80
10	3500	200	9.87	0.08	0.00	0.00
				3433	1868	12
10	7000	50	22.13	0.19	0.02	0.01
				3183	4267	98
10	7000	100	20.81	0.26	0.02	0.01
				3420	3679	84
10	7000	200	17.35	0.35	0.09	0.02
				3238	4718	310
11	1750	50	2.75	0.02	0.01	0.01
				7953	2769	29
11	1750	100	2.61	0.02	0.00	0.00
				7958	2331	12
11	1750	200	2.26	0.01	0.00	0.00
				7880	2233	7
11	3500	50	5.25	0.06	0.02	0.01
				7691	6675	291
11	3500	100	4.99	0.04	0.00	0.00
				7769	5177	125
11	3500	200	4.36	0.03	0.00	0.00
				7911	3204	24
11	7000	50	9.57	0.16	0.02	0.02
				7179	12042	280
11	7000	100	9.21	0.16	0.02	0.02
				7437	9968	223
11	7000	200	8.17	0.11	0.01	0.01
				7656	7452	71

Table 3: Results for the root node ($K = 3$)

T	β/h	α/h	LP Gap %	$\|\mathcal{Q}\| = 1$	$\|\mathcal{Q}\| = 2$	General $\mathcal{Q}$
6	1750	50	10.03	0.62	0.04	0.03
				1560	3243	98
6	1750	100	8.26	0.65	0.06	0.04
				1479	4139	144
6	1750	200	5.36	0.54	0.02	0.01
				1438	5784	33
6	3500	50	16.29	1.29	0.28	0.19
				1464	6553	311
6	3500	100	13.76	1.24	0.21	0.17
				1442	6939	120
6	3500	200	9.39	0.95	0.06	0.05
				1436	7412	78
6	7000	50	23.52	1.97	0.38	0.27
				1365	10041	334
6	7000	100	20.93	2.18	0.40	0.31
				1422	10044	335
6	7000	200	15.59	1.81	0.24	0.17
				1405	12248	183
7	1750	50	4.90	0.28	0.04	0.03
				5706	9580	423
7	1750	100	4.38	0.33	0.03	0.02
				5524	12058	298
7	1750	200	3.32	0.28	0.01	0.01
				5341	15223	77
7	3500	50	8.51	0.55	0.08	0.06
				5434	19017	894
7	3500	100	7.75	0.55	0.06	0.05
				5384	20521	466
7	3500	200	6.12	0.52	0.04	0.03
				5335	21474	361
7	7000	50	14.04	0.75	0.16	0.13
				5147	26233	588
7	7000	100	13.03	0.82	0.16	0.13
				5184	28916	590
7	7000	200	10.64	0.85	0.13	0.10
				5197	29711	592

Table 4: Results for the root node ($K = 4$)

T	β/h	α/h	LP Gap %	$\lvert\mathcal{Q}\rvert = 1$	$\lvert\mathcal{Q}\rvert = 2$	General $\mathcal{Q}$
5	1750	50	8.80	1.35	0.21	0.17
				1905	7381	133
5	1750	100	7.42	1.25	0.15	0.08
				1894	7347	213
5	1750	200	4.66	1.47	0.09	0.08
				1651	18741	61
5	3500	50	13.12	1.68	0.27	0.21
				1852	10956	215
5	3500	100	11.40	1.88	0.29	0.20
				1842	12182	369
5	3500	200	7.52	2.33	0.30	0.22
				1619	21298	321
5	7000	50	14.06	1.53	0.33	0.24
				1781	13067	1838
5	7000	100	17.32	3.36	0.75	0.60
				1679	18449	341
5	7000	200	12.10	3.28	0.71	0.52
				1546	32367	477
6	1750	50	4.25	0.53	0.07	0.05
				9779	28553	797
6	1750	100	3.73	0.66	0.08	0.06
				9310	53983	904
6	1750	200	2.92	0.69	0.05	0.04
				8561	70253	336
6	3500	50	7.17	0.88	0.17	0.12
				9380	65631	1438
6	3500	100	6.41	1.05	0.20	0.16
				8979	75479	1318
6	3500	200	5.24	1.12	0.15	0.12
				8487	74747	645
6	7000	50	11.20	1.32	0.35	0.27
				8589	89049	1658
6	7000	100	10.31	1.55	0.45	0.39
				8339	93640	1160
6	7000	200	8.84	1.62	0.42	0.35
				8383	98949	1358

Table 5: Results for branch-and-cut ($K = 2$)

T	β/h	α/h	No. of cuts	Optimality gap %	Nodes	CPU secs
10	1750	50	519	0.00	1239	4.4
			4676	0.00	0	0.7
10	1750	100	505	0.00	103	1.6
			4749	0.00	0	0.6
10	1750	200	464	0.00	4	0.7
			4575	0.00	0	0.5
10	3500	50	612	0.00	131850	220.2
			5996	0.00	0	3.0
10	3500	100	598	0.00	39828	70.8
			6129	0.00	0	5.4
10	3500	200	513	0.00	343	2.4
			5313	0.00	0	1.8
10	7000	50	671	0.00	1336827	2619.7
			7737	0.00	0	13.9
10	7000	100	682	0.00	915006	1715.7
			7213	0.00	0	5.0
10	7000	200	597	0.00	13124	26.0
			8407	0.00	0	23.5
11	1750	50	882	0.00	30	2.5
			10751	0.00	0	1.7
11	1750	100	859	0.00	3	1.9
			10301	0.00	0	1.6
11	1750	200	780	0.00	3	1.2
			10120	0.00	0	1.6
11	3500	50	1065	0.00	644407	820.2
			14946	0.00	0	63.5
11	3500	100	994	0.00	9807	42.9
			13071	0.00	0	3.3
11	3500	200	852	0.00	889	9.2
			11139	0.00	0	2.5
11	7000	50	1126	0.03[3]	826644	***
			20784	0.00	0	189.0
11	7000	100	1112	0.03[3]	907471	***
			17796	0.00	0	35.9
11	7000	200	1084	0.00	414122	1496.7
			15179	0.00	0	15.5

Table 6: Results for branch-and-cut ($K = 3$)

T	β/h	α/h	No. of cuts	Optimality gap %	Nodes	CPU secs
6	1750	50	523	0.01[1]	1010894	60.1
			4957	0.00	0	3.0
6	1750	100	551	0.00	157889	244.9
			5896	0.00	4	9.2
6	1750	200	489	0.00	4913	9.1
			7259	0.00	0	1.5
6	3500	50	575	0.12[3]	2703911	***
			9507	0.00	373	91.7
6	3500	100	573	0.14[3]	2787691	***
			9618	0.00	438	131.9
6	3500	200	540	0.00	253920	387.6
			9091	0.00	20	11.2
6	7000	50	507	0.23[3]	2879642	***
			13746	0.00	9409	2207.5
6	7000	100	528	0.39[3]	3154270	***
			14552	0.05[2]	8356	867.3
6	7000	200	609	0.57[3]	2777630	***
			15072	0.02[2]	5533	90.1
7	1750	50	1236	0.09[3]	1148262	***
			15971	0.00	0	31.7
7	1750	100	1220	0.07[3]	1181449	***
			18187	0.00	13	85.1
7	1750	200	1117	0.02[3]	967725	***
			20653	0.00	0	19.3
7	3500	50	1306	0.21[3]	1076628	***
			28354	0.00	2751	3218.1
7	3500	100	1300	0.17[3]	1089148	***
			27531	0.00	286	724.6
7	3500	200	1209	0.10[3]	1059317	***
			27589	0.00	0	143.4
7	7000	50	1255	0.31[3]	1045952	***
			35932	0.02[1]	2172	3078.9
7	7000	100	1340	0.29[3]	1004477	***
			37756	0.02[3]	2000	***
7	7000	200	1332	0.27[3]	1085362	***
			38215	0.02[3]	1768	***

Table 7: Results for branch-and-cut ($K = 4$)

T	β/h	α/h	No. of cuts	Optimality gap %	Nodes	CPU secs
5	1750	50	670	0.12[3]	2185170	***
			10158	0.00	251	59.1
5	1750	100	660	0.03[3]	1925658	***
			9585	0.00	47	24.4
5	1750	200	575	0.09[2]	1858810	1506.3
			20931	0.00	24	98.2
5	3500	50	694	0.10[3]	1997388	***
			13399	0.00	1794	356.7
5	3500	100	716	0.15[3]	2257218	***
			14643	0.00	208	99.7
5	3500	200	673	0.21[3]	2174847	***
			24571	0.00	480	636.9
5	7000	50	642	0.04[2]	1175516	213.9
			18065	0.00	806	1275.2
5	7000	100	858	0.37[3]	1570320	***
			25026	0.10[2]	2057	3451.2
5	7000	200	620	0.33[3]	2009171	***
			36770	0.07[2]	600	993.6
6	1750	50	2071	0.22[3]	658145	***
			40204	0.00	155	817.5
6	1750	100	2043	0.24[3]	643715	***
			67106	0.01[3]	483	***
6	1750	200	1810	0.17[3]	708248	***
			80495	0.00	198	2003.2
6	3500	50	1984	0.42[3]	633599	***
			79711	0.05[3]	425	***
6	3500	100	1987	0.47[3]	619146	***
			88734	0.07[3]	143	***
6	3500	200	1973	0.37[3]	630579	***
			85886	0.04[3]	112	***
6	7000	50	1771	0.67[3]	611857	***
			102151	0.14[3]	46	***
6	7000	100	2048	0.72[3]	617064	***
			105606	0.24[3]	0	***
6	7000	200	2022	0.57[3]	634604	***
			112756	0.24[3]	0	***

CHAPTER 5

SEQUENTIAL PAIRING OF MIXED INTEGER INEQUALITIES

5.1 Introduction

Motivated by the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequalities developed in Chapter 4, in this chapter, we develop a scheme for generating new valid inequalities for mixed integer programs by taking pairwise combinations of existing valid inequalities. Our scheme is related to the mixed integer rounding (MIR) procedure of Nemhauser and Wolsey [76, 77], the mixing procedure of Günlük and Pochet [48], and the split cuts of Cook, Kannan and Schrijver [28]. We derive new inequalities iteratively by a simple combination of two inequalities at a time, which we call *pairing*. As will be seen, the order in which the inequalities are paired is important since the resulting new inequalities depend on the order.

We describe the pairing procedure for pure integer programs and present a simple extension to MIPs in the next section. We study two structures in Sections 5.3 and 5.4 for which our pairing procedure gives nice results. We say that a set of inequalities is *nested* if component by component the coefficients in each successive inequality are no smaller than the coefficients in the previous inequalities. In the nested case, we show that there is a unique order for combining the inequalities that gives all of the nondominated inequalities that can be generated by the procedure. In this case, we obtain only a small number of inequalities and separation is fast. Moreover, we provide sufficient conditions for which the resulting inequalities are facet-defining. We say that a set of inequalities is *disjoint* if each integer variable appears in only one of the inequalities. Such disjoint sets arise in two-stage stochastic integer programming. Here we are again able to characterize the nondominated inequalities generated by the procedure, and we give a polynomial time separation algorithm. We also provide sufficient facet-defining conditions.

Section 5.5 focuses on some applications of our procedure. In Section 5.6, we present computational results for nested and disjoint sets to demonstrate the strength of the inequalities in improving linear programming relaxation bounds. Final remarks are presented in Section 5.7. The results of this chapter also appear in [46].

5.2 The Pairing Scheme

Given a set of non-negative integer vectors $Y \subset \mathbb{Z}^n_+$, a vector $a \in \mathbb{R}^{n+1}$ *defines* a valid inequality for Y if

$$\sum_{j=1}^{n} a_j y_j - a_{n+1} \geq 0 \quad \text{for all } y \in Y.$$

Given two such valid inequalities defined by vectors a and b, the one defined by a *dominates* the one defined by b if $a_j \leq b_j$ for all $j = 1, \ldots, n$ and $a_{n+1} \geq b_{n+1}$. We write $a \succeq b$.

The inequality $a \leq b$ for two vectors a and b of the same dimension is meant to hold component-wise. Similarly, $\min(a, b)$ and $\max(a, b)$ is understood to be carried out component-wise. For brevity, given a vector a and a scalar γ, we define $a + \gamma = a + \gamma \mathbb{1}$ and $\min\{a, \gamma\} = \min\{a, \gamma \mathbb{1}\}$, where $\mathbb{1}$ is a vector of ones of the same dimension as a.

Definition 5.1 *Given $a, b \in \mathbb{R}^{n+1}$ with $b_{n+1} \geq a_{n+1}$, we define the pairing of a and b as*

$$a \circ b = \min\{a + b_{n+1} - a_{n+1}, \max(a, b)\},$$

i.e., $(a \circ b)_{n+1} = b_{n+1}$ and

$$(a \circ b)_j = \begin{cases} a_j & \text{if } a_j \geq b_j \\ b_j & \text{if } a_j \leq b_j,\ b_j \leq a_j + b_{n+1} - a_{n+1} \\ a_j + b_{n+1} - a_{n+1} & \text{if } a_j \leq b_j,\ b_j \geq a_j + b_{n+1} - a_{n+1}, \end{cases}$$

for all $j = 1, \ldots, n$.

Theorem 5.1 *If $a, b \in \mathbb{R}^{n+1}$ define two valid inequalities for Y, then $a \circ b$ defines a valid inequality for Y.*

Proof: Without loss of generality, assume that $b_{n+1} \geq a_{n+1}$. Thus $(a \circ b)_{n+1} = b_{n+1}$. Then, given $y \in Y$, we need to show that

$$\sum_{j=1}^{n} (a \circ b)_j y_j \geq b_{n+1}. \tag{5.1}$$

Let $J = \{j \in \{1, \ldots, n\} : a_j + b_{n+1} - a_{n+1} < \max(a_j, b_j)\}$ and $\overline{J} = \{1, \ldots, n\} \setminus J$. Then the left-hand side of (5.1) can be written as

$$\sum_{j \in J} a_j y_j + \sum_{j \in \overline{J}} \max(a_j, b_j) y_j + (b_{n+1} - a_{n+1}) \sum_{j \in J} y_j. \tag{5.2}$$

If there exists $j^* \in J$ such that $y_{j^*} \geq 1$, then (5.2) is

$$\geq \sum_{j=1}^{n} a_j y_j + (b_{n+1} - a_{n+1}) \geq b_{n+1},$$

where the last inequality follows from the validity of the inequality defined by a. On the other hand, if $y_j = 0$ for all $j \in J$, then (5.2) is

$$\geq \sum_{j \in J} b_j y_j + \sum_{j \in \overline{J}} b_j y_j \geq b_{n+1},$$

where the last inequality follows from the validity of the inequality defined by b. Thus $a \circ b$ defines a valid inequality for Y. □

In addition to the above simple and direct proof of Theorem 5.1, there is a proof that uses the MIR procedure [76], another proof that uses split cuts [28] and a third proof, for the case of nonnegative coefficients, that follows from Günlük and Pochet mixing [48].

Example. Consider the set

$$Y = \Big\{ y \in \mathbb{Z}^3_+ : 3y_1 + 5y_2 \geq 3,\ 5y_2 + 4y_3 \geq 5 \Big\}.$$

The two original inequalities for Y are defined by $a = (3, 5, 0, 3)$ and $b = (0, 5, 4, 5)$. The valid inequality defined by $a \circ b$ is

$$3y_1 + 5y_2 + 2y_3 \geq 5. \tag{5.3}$$

To see that (5.3) can be useful, note that it cuts off the fractional point $(0, 3/5, 1/2)$ which is feasible to the LP relaxation of Y.

The pairing scheme can be easily applied to mixed-integer sets. The pair $(a, g) \in \mathbb{R}^{n+1} \times \mathbb{R}^p$, defines a valid inequality for a mixed-integer set $X \subset \mathbb{Z}^n_+ \times \mathbb{R}^p_+$ if

$$\sum_{i=1}^{n} a_i y_i + \sum_{j=1}^{p} g_j x_j \geq a_{n+1} \quad \text{for all } (y, x) \in X.$$

Corollary 5.1 *If (a^1, g^1) and (a^2, g^2) define two valid inequalities for X, then $(a^1 \circ a^2, \max\{ g^1, g^2\})$ defines a valid inequality for X.*

Note that the standard disjunctive inequality (see, e.g. [76]), obtained from the inequalities (a^1, g^1) and (a^2, g^2) for X,

$$\sum_{i=1}^{n} \max\{a_i^1, a_i^2\} y_i + \sum_{j=1}^{p} \max\{g_j^1, g_j^2\} x_j \geq \min\{a_{n+1}^1, a_{n+1}^2\},$$

is dominated by the pairing inequality in Corollary 1.

We now consider the pairing inequalities obtained from a set of inequalities. Suppose we have K valid inequalities for Y defined by the vectors $\{a^1, \ldots, a^K\} \subset \mathbb{R}^{n+1}$. Given a subset of these K vectors, we can obtain new valid inequalities by carrying out a sequence of pairing operations. For example, the valid inequality defined by the vector $((a^{k_1} \circ a^{k_2}) \circ (a^{k_2} \circ a^{k_3})) \circ a^{k_4}$ is obtained from $\{a^{k_1}, a^{k_2}, a^{k_3}, a^{k_4}\}$ with the parentheses distinguishing the sequence in which the pairings are carried out. Since the $\circ$ operation is not associative, the valid inequalities obtained from a given set of vectors *depends on the sequence* in which the pairings are done. Thus from the set of K valid inequalities defined by $\{a^1, \ldots, a^K\}$ we can generate an exponential number of inequalities depending on the subset of valid inequalities chosen and the sequence in which they are mixed. A key problem is to identify pairing sequences that lead to good sets of valid inequalities, i.e., strong inequalities over which separation can be done efficiently.

In the following two sections, we investigate a pairing sequence that leads to two such families of inequalities. This pairing sequence is defined by

Definition 5.2 *Given a finite set of vectors, i.e., $A = \{a^1, \ldots, a^K\}$, where $a_{n+1}^1 \leq a_{n+1}^2 \leq \ldots \leq a_{n+1}^K$, we define sequential pairing of the vectors in A by*

$$\Delta(A) = ((\ldots((a^1 \circ a^2) \circ a^3) \circ \ldots) \circ a^K).$$

5.3 The Nested Case

Consider a set $A = \{a^1, \ldots, a^K\} \subset \mathbb{R}^{n+1}$ such that $a^1 \leq \cdots \leq a^K$. Then, we say that the valid inequalities defined by the vectors in A are (or the set A itself is) *nested*. Here we consider mixed integer systems where the coefficients of the integer variables are nested. Nested sets arise, for example, in the dynamic knapsack problem considered by Loparic, Marchand and Wolsey [67] where the feasible region is given by

$$X = \Big\{(y,x) \in \{0,1\}^n \times \mathbb{R}_+ : \quad \sum_{j=1}^{i} a_j y_j + x \geq \sum_{j=1}^{i} d_j, \quad i = 1,\ldots,n\Big\}, \tag{5.4}$$

with $a \in \mathbb{R}^n_+$ and $d \in \mathbb{R}^n_+$. Here, x is a continuous inventory variable, $y_j \in \{0,1\}$ represents whether the amount a_j is produced in period j, and d_j is the demand in period j.

Let $A_k = \{a^1,\ldots,a^k\}$ for $k = 1,\ldots,K$, and let $\Phi(A) \in \mathbb{R}^{n+1}$ be a vector obtained by an arbitrary sequence of pairings of the vectors in A. Next, we show that $\Delta(A) \succeq \Phi(A)$.

Theorem 5.2 *If $A = \{a^1,\ldots,a^K\}$ is nested, then*

$$\Delta(A) = \min\{a^1 + a^K_{n+1} - a^1_{n+1}, a^2 + a^K_{n+1} - a^2_{n+1}, \ldots, a^{K-1} + a^K_{n+1} - a^{K-1}_{n+1}, a^K\}.$$

Proof: The proof is by induction. For $K = 2$, we have $a^1 \leq a^2$, then

$$\begin{aligned}\Delta(A_2) &= \min\{a^1 + a^2_{n+1} - a^1_{n+1}, \max\{a^1, a^2\}\} \\ &= \min\{a^1 + a^2_{n+1} - a^1_{n+1}, a^2\}.\end{aligned}$$

Assume that the claim holds for $K = k$, i.e., $\Delta(A_k) = \min\{a^1 + a^k_{n+1} - a^1_{n+1}, a^2 + a^k_{n+1} - a^2_{n+1}, \ldots, a^{k-1} + a^k_{n+1} - a^{k-1}_{n+1}, a^k\}$. Then

$$\begin{aligned}\Delta(A_{k+1}) &= \Delta(A_k) \circ a^{k+1} \\ &= \min\{\Delta(A_k) + a^{k+1}_{n+1} - a^k_{n+1}, \max\{\Delta(A_k), a^{k+1}\}\} \\ &= \min\{\Delta(A_k) + a^{k+1}_{n+1} - a^k_{n+1}, a^{k+1}\} \\ &= \min\{a^1 + a^{k+1}_{n+1} - a^1_{n+1}, \ldots, a^{k-1} + a^k_{n+1} - a^{k-1}_{n+1}, \\ &\qquad a^k + a^{k+1}_{n+1} - a^k_{n+1}, a^{k+1}\},\end{aligned}$$

where the third equality follows from the fact that $\Delta(A_k) \leq a^k \leq a^{k+1}$. Thus the claim holds. $\square$

Lemma 5.1 *If $A = \{a^1, \ldots, a^K\}$ and $B = \{b^1, \ldots, b^R\}$ are nested sets such that $A \cup B = \{a^1, \ldots, a^K, b^1, \ldots, b^B\}$ is nested, then*

$$\Delta(A \cup B) \succeq \Delta(A) \circ \Delta(B).$$

Proof: Since $\Delta(A \cup B)_{n+1} = (\Delta(A) \circ \Delta(B))_{n+1}$, it is sufficient to show that $\Delta(A \cup B) \leq \Delta(A) \circ \Delta(B)$. We have

$$\begin{aligned}
\Delta(A \cup B) &= \min\{a^1 + b^R_{n+1} - a^1_{n+1}, \ldots, a^K + b^R_{n+1} - a^K_{n+1}, \\
&\qquad b^1 + b^R_{n+1} - b^1_{n+1}, \ldots, b^R\} \\
&= \min\{a^1 + a^K_{n+1} - a^1_{n+1} + (b^R_{n+1} - a^K_{n+1}), \ldots, a^K + (b^R_{n+1} - a^K_{n+1}), \\
&\qquad \min\{b^1 + b^R_{n+1} - b^1_{n+1}, \ldots, b^R\}\} \\
&= \min\{\Delta(A) + b^R_{n+1} - a^K_{n+1}, \Delta(B)\} \\
&\leq \min\{\Delta(A) + b^R_{n+1} - a^K_{n+1}, \max\{\Delta(A), \Delta(B)\}\} \\
&= \Delta(A) \circ \Delta(B).
\end{aligned}$$

□

Lemma 5.2 *If $a, b, c, d \in \mathbb{R}^{n+1}$ are such that $a \succeq c$, $b \succeq d$, $a_{n+1} = c_{n+1}$ and $b_{n+1} = d_{n+1}$, then $a \circ b \succeq c \circ d$.*

Proof: Without loss of generality, assume that $d_{n+1} = b_{n+1} \geq a_{n+1} = c_{n+1}$. Then

$$(a \circ b)_{n+1} = b_{n+1} = d_{n+1} = (c \circ d)_{n+1}.$$

Since $a \succeq c$ and $b \succeq d$, we have $\max(a_j, b_j) \leq \max(c_j, d_j)$ for all $j = 1, \ldots, n$; and since $b_{n+1} = d_{n+1}$, $a_{n+1} = c_{n+1}$ and $a_j \leq c_j$ for all $j = 1, \ldots, n$, we have $a_j + b_{n+1} - a_{n+1} = a_j + d_{n+1} - c_{n+1} \leq c_j + d_{n+1} - c_{n+1}$ for all $j = 1, \ldots, n$. Thus

$$\begin{aligned}
(a \circ b)_j &= \min\{a_j + b_{n+1} - a_{n+1}, \max(a_j, b_j)\} \\
&\leq \min\{c_j + d_{n+1} - c_{n+1}, \max(c_j, d_j)\} \\
&= (c \circ d)_j
\end{aligned}$$

for all $j = 1, \ldots, n$. The claim then follows from the definition of $\succeq$. $\square$

Theorem 5.3 *If A is nested, then $\Delta(A) \succeq \Phi(A)$ for any $\Phi(A)$.*

Proof: The proof is by induction on $|A|$. Note that the claim holds trivially for nested sets A such that $|A| \leq 2$. Assume that the claim holds for all nested sets A such that $|A| \leq k$.

Consider a nested set A such that $|A| = k + 1$. Given $\Phi(A)$, obtained by an arbitrary sequence of pairings of the vectors in A, we can write

$$\Phi(A) = \Phi(A^1) \circ \Phi(A^2)$$

for some $A^1, A^2 \subset A$ such that $A^1 \cap A^2 = \emptyset$ and $A^1 \cup A^2 = A$. Note that $|A^1| \leq k$ and $|A^2| \leq k$. Thus by our induction hypothesis $\Delta(A^1) \succeq \Phi(A^1)$ and $\Delta(A^2) \succeq \Phi(A^2)$. We also notice that $\Delta(A^2)_{n+1} = \Phi(A^2)_{n+1}$ and $\Delta(A^1)_{n+1} = \Phi(A^1)_{n+1}$. Then

$$\begin{aligned} \Phi(A) &\preceq \Delta(A^1) \circ \Delta(A^2) \\ &\preceq \Delta(A^1 \cup A^2) = \Delta(A), \end{aligned}$$

where the first statement follows from Lemma 5.2 and the second statement follows from Lemma 5.1. $\square$

Lemma 5.3 *If $A = \{a^1, \ldots, a^K\}$ is nested and $B \subset A$ is such that $a^K \in B$, then $\Delta(A) \succeq \Delta(B)$.*

Proof: Since $\Delta(B)_{n+1} = \Delta(A)_{n+1}$, it is sufficient to show that $\Delta(A) \leq \Delta(B)$. Let $A \setminus B = \{a^{i_1}, \ldots, a^{i_l}\}$ and $B = \{a^{j_1}, \ldots, a^{j_m}, a^K\}$. Then

$$\begin{aligned} \Delta(A) &= \min\{a^{i_1} + a^K_{n+1} - a^{i_1}_{n+1}, \ldots, a^{i_l} + a^K_{n+1} - a^{i_l}_{n+1}, \\ &\qquad a^{j_1} + a^K_{n+1} - a^{j_1}_{n+1}, \ldots, a^{j_m} + a^K_{n+1} - a^{j_m}_{n+1}, a^K\} \\ &= \min\{a^{i_1} + a^K_{n+1} - a^{i_1}_{n+1}, \ldots, a^{i_l} + a^K_{n+1} - a^{i_l}_{n+1}, \Delta(B)\} \\ &\leq \Delta(B). \end{aligned}$$

$\square$

Combining Theorem 5.3 and Lemma 5.3, we obtain

Theorem 5.4 *Let $A = \{a^1, \ldots, a^K\}$ be nested. All the non-dominated inequalities obtained by pairings of the vectors in A are contained in the set* $\cup_{k=1}^{K} \{\Delta(A_k)\}$.

Hence there are at most K non-dominated inequalities.

Now we give sufficient conditions for the inequalities in $\cup_{k=1}^{K}\Delta(A_k)$ to be facet-defining for a particular class of nested systems. Let $A = \{a^1, \ldots, a^K\} \in \mathbb{R}^{n+1}$ be a nested set such that $a^i \geq 0$ for all $i = 1, \ldots, K$, and consider the mixed 0-1 set (with one continuous variable):

$$X = \Big\{(y, x) \in \{0,1\}^n \times \mathbb{R}_+ : \ \sum_{j=1}^{n} a^i_j y_j + x \geq a^i_{n+1}, \ i = 1, \ldots, K\Big\}.$$

Without loss of generality, we assume that $a^i_j \leq a^i_{n+1}$ for all $j = 1, \ldots, n$ and $i = 1, \ldots, K$, since otherwise the coefficients can be strengthened to $a^i_j = a^i_{n+1}$. Let $A_i = \{a^1, \ldots, a^i\}$ for $i = 1, \ldots, K$, and $\Delta^i = \Delta(A_i)$.

Theorem 5.5 *Given $i \in \{1, \ldots, K\}$, the sequential pairing inequality*

$$\sum_{j=1}^{n} \Delta^i_j y_j + x \geq a^i_{n+1} \tag{5.5}$$

is facet-defining for conv(X) if, for all $k \in \{i, i+1, \ldots, K\}$,

(a) there exists $j^* \in \{1, \ldots, n\}$ such that $\Delta^i_{j^*} + a^k_{n+1} - a^i_{n+1} \leq a^k_{j^*}$, and

(b) $\sum_{j \in Z(i)} a^k_j \geq a^k_{n+1} - a^i_{n+1}$ where $Z(i) = \{j \in \{1, \ldots, n\} : a^i_j = 0\}$.

Proof: We construct $\dim(X) = n + 1$ linearly independent vectors belonging to X that satisfy (5.5) at equality.

We construct a vector corresponding to each of the $n+1$ variables. Let e^x and e^{y_j} be unit vectors in $\mathbb{R}^{n+1}$ corresponding to the coordinates x and y_j for $j = 1, \ldots, n$. The constructed vectors are denoted by $\{u^j\}_{j=0}^{n}$ and are constructed as follows.

(i) Vector u^0 corresponds to variable x and is given by

$$u^0 = a^i_{n+1} e^x + \sum_{r \in Z(i)} e^{y_r}.$$

(ii) Vector u^{j^*} corresponds to variable y_{j^*} and is given by

$$u^{j^*} = [a^i_{n+1} - \Delta^i_{j^*}] e^x + e^{y_{j^*}}.$$

(iii) For each y_j where $j \in \{1, \ldots, n\} \setminus \{Z(i) \cup \{j^*\}\}$, the corresponding vector u^j is given by

$$u^j = [a^i_{n+1} - \Delta^i_j] e^x + e^{y_j} + \sum_{r \in Z(i)} e^{y_r}.$$

Note that there are $n - |Z(i)| - 1$ such vectors.

(iv) For each y_j where $j \in Z(i)$, the corresponding vector u^j is given by

$$u^j = [a^i_{n+1} - \Delta^i_{j^*}] e^x + e^{y_j} + e^{y_{j^*}}.$$

Feasibility: We need to show that $\{u^j\}_{j=0}^n$ satisfies

$$a^k u \geq a^k_{n+1} \qquad k = 1, \ldots, K, \tag{5.6}$$

$$u_0 \geq 0,\ u_j \in \{0, 1\} \qquad j = 1, \ldots, n, \tag{5.7}$$

where $a^k_0 = 1$ for all $k = 1, \ldots, K$.

(i) The vector u^0 clearly satisfies (5.7) since $a^i_{n+1} \geq 0$. The left-hand-side of (5.6) is

$$\begin{aligned} a^k u^0 &= a^i_{n+1} + \textstyle\sum_{j \in Z(i)} a^k_j \\ &\geq \begin{cases} a^k_{n+1} & \text{if } k < i \\ a^i_{n+1} + a^k_{n+1} - a^i_{n+1} = a^k_{n+1} & \text{if } k \geq i, \end{cases} \end{aligned}$$

where the inequality for the case $k < i$ follows from the fact that $a^k_j \geq 0$ for all k, j, and $a^k_{n+1} \leq a^i_{n+1}$ for all $k < i$; and the inequality for the case $k \geq i$ follows from condition (b) of the Theorem. Thus u^0 satisfies (5.6).

(ii) The vector u^{j^*} clearly satisfies (5.7) since $\Delta^i_{j^*} \leq a^i_{j^*} \leq a^i_{n+1}$. The left-hand-side of (5.6) corresponding to $k < i$ is

$$\begin{aligned} a^k u^{j^*} &= a^i_{n+1} - \Delta^i_{j^*} + a^k_{j^*} \\ &\geq a^i_{n+1} - (a^k_{j*} + a^i_{n+1} - a^k_{n+1}) + a^k_{j^*} = a^k_{n+1}, \end{aligned}$$

where the inequality follows from the fact that $\Delta^i_{j*} \leq a^k_{j*} + a^i_{n+1} - a^k_{n+1}$ for all $k = 1, \ldots, i$. The left-hand-side of (5.6) corresponding to $k \geq i$ is

$$\begin{aligned} a^k u^{j^*} &= a^i_{n+1} - \Delta^i_{j^*} + a^k_{j^*} \\ &\geq a^k_{n+1} - a^i_{n+1} + a^i_{n+1} = a^k_{n+1}, \end{aligned}$$

where the inequality follows from condition (a). Thus u^{j^*} satisfies (5.6).

(iii) For a given $j \in \{1, \ldots, n\} \setminus \{Z(i) \cup \{j^*\}\}$, the vector u^j clearly satisfies (5.7) since $\Delta^i_j \leq a^i_j \leq a^i_{n+1}$. The left-hand-side of (5.6) corresponding to $k < i$ is

$$\begin{aligned} a^k u^j &= a^i_{n+1} - \Delta^i_j + a^k_j + \textstyle\sum_{r \in Z(i)} a^k_r \\ &= a^i_{n+1} - \Delta^i_j + a^k_j \\ &\geq a^i_{n+1} - (a^k_j + a^i_{n+1} - a^k_{n+1}) + a^k_j = a^k_{n+1}, \end{aligned}$$

where the second line follows from the nested property $a^k_r \leq a^i_r$ for all $k = 1, \ldots, i$, $r = 1, \ldots, n$, and $a^i_r = 0$ for all $r \in Z(i)$; and the third line follows from the fact that $\Delta^i_j \leq a^k_j + a^i_{n+1} - a^k_{n+1}$ for all $k = 1, \ldots, i$. The left-hand-side of (5.6) corresponding to $k \geq i$ is

$$\begin{aligned} a^k u^j &= a^i_{n+1} - \Delta^i_j + a^k_j + \textstyle\sum_{r \in Z(i)} a^k_r \\ &\geq a^i_{n+1} - \Delta^i_j + a^k_j + a^k_{n+1} - a^i_{n+1} \\ &= -\Delta^i_j + a^k_j + a^k_{n+1} \\ &\geq a^k_{n+1}, \end{aligned}$$

where the second line follows from condition (b), and the last line follows from the fact that $a^k_j \geq a^i_j \geq \Delta^i_j$ for all $k = i, i+1, \ldots, K$. Thus u^j satisfies (5.6).

(iv) For a given $j \in Z(i)$ the vector u^j clearly satisfies (5.7) since $\Delta^i_{j^*} \leq a^i_{j^*} \leq a^i_{n+1}$. The vector u^j also satisfies (5.6) since $u^j \geq u^{j*}$ and u^{j^*} satisfies (5.6).

Tightness: It is easily verified that the vectors $\{u^j\}_{j=0}^n$ satisfy the inequality (5.5) as an equality.

Linear independence: To verify the linear independence of the $n+1$ vectors $\{u^j\}_{j=0}^n$, observe that we can obtain $n+1$ unit vectors from $\{u^j\}_{j=0}^n$ as follows:

$e^{y_j} = u^j - u^{j^*}$ for all $j \in Z(i)$.

$e^x = u^0 - \sum_{j \in Z(i)} e^{y_j}$.

$e^{y_{j^*}} = u^{j^*} - [a^i_{n+1} - \Delta^i_{j^*}]e^x$.

$e^{y_j} = u^j - [a^i_{n+1} - \Delta^i_j]e^x - \sum_{r \in Z(i)} e^{y_r}$ for all $j \in \{1, \ldots, n\} \setminus \{Z(i) \cup \{j^*\}\}$.

□

5.4 The Disjoint Case

A set $A = \{a^1, \ldots, a^K\} \subset \mathbb{R}^{n+1}$ satisfying

(i) $a^k \geq 0$ for all $k = 1, \ldots, K$,

(ii) for any two vectors a^l and a^m, $a^l_j a^m_j = 0$ for $j = 1, \ldots, n$, and

(iii) $a^1_{n+1} \leq a^2_{n+1} \leq \cdots \leq a^K_{n+1}$.

is said to be *disjoint*. Here we consider mixed integer systems where the coefficients of the integer variables are disjoint. An example is the deterministic equivalent formulation of a two-stage stochastic program with integer second stage variables [19]

$$\begin{aligned} \min \quad & c^T x + \textstyle\sum_{s=1}^S p_s q_s^T y_s \\ & x \in X \subseteq \mathbb{R}_+^{n_1 - p_1} \times \mathbb{Z}_+^{p_1} \\ & T_s x + W_s y_s \geq h_s \qquad s = 1, \ldots, S \\ & y_s \in \mathbb{Z}_+^{n_2} \qquad\qquad s = 1, \ldots, S. \end{aligned} \tag{5.8}$$

In (5.8), there are two sets of decision variables. The first-stage variables x are decided prior to a scenario s of realizations of the uncertain problem parameters (q_s, T_s, W_s, h_s). The second-stage decisions y_s constitute "recourse" actions corresponding to the scenario s realized. A scenario s occurs with probability p_s, and the objective is to minimize the sum of first-stage and expected second-stage costs. Note that the second-stage variables constitute a disjoint system.

Theorem 5.6 *If $A = \{a^1, \ldots, a^K\}$ is disjoint, then*

$$\Delta(A) = a^1 + \sum_{i=2}^{K} \min\{a^i_{n+1} - a^{i-1}_{n+1}, a^i\}.$$

Proof: The proof follows directly from Definitions 5.1 and 5.2, and the definition of a disjoint set. □

Lemma 5.4 *Let $A = \{a^1, \ldots, a^K\}$ and $B = \{b^1, \ldots, b^R\}$ be disjoint sets such that $A \cup B$ is disjoint and $a^K_{n+1} \leq b^R_{n+1}$. Then there exists $C \subseteq A \cup B$ with $b^R \in C$ such that*

$$\Delta(C) \succeq \Delta(A) \circ \Delta(B).$$

Proof: Since $\Delta(C)_{n+1} = b^R_{n+1} = (\Delta(A) \circ \Delta(B))_{n+1}$, it is sufficient to show that $\Delta(C) \leq \Delta(A) \circ \Delta(B)$. From Theorem 5.6, we have

$$\begin{aligned}
\Delta(A) \circ \Delta(B) &= (a^1 + \sum_{i=2}^{K} \min\{a^i_{n+1} - a^{i-1}_{n+1}, a^i\}) \circ (b^1 + \sum_{i=2}^{R} \min\{b^i_{n+1} - b^{i-1}_{n+1}, b^i\}) \\
&= (a^1 + \sum_{i=2}^{K} \min\{a^i_{n+1} - a^{i-1}_{n+1}, a^i\}) \\
&\quad \underbrace{+ \min\{b^R_{n+1} - a^K_{n+1}, (b^1 + \sum_{i=2}^{R} \min\{b^i_{n+1} - b^{i-1}_{n+1}, b^i\})\}}_{(d^{AB})}.
\end{aligned}$$

Let $i^* = \min\{i \in \{1,\ldots,R\} : \ b^i_{n+1} \geq a^K_{n+1}\}$ and $C = A \cup \{b^{i^*},\ldots,b^R\}$. Note that C is disjoint. Then

$$\begin{aligned}\Delta(C) &= a^1 + \sum_{i=2}^{K} \min\{a^i_{n+1} - a^{i-1}_{n+1}, a^i\} \\ & \quad \underbrace{+ \min\{b^{i^*}_{n+1} - a^K_{n+1}, b^{i^*}\} + \sum_{i=i^*+1}^{R} \min\{b^i_{n+1} - b^{i-1}_{n+1}, b^i\}}_{(d^C)}\end{aligned}$$

By letting $b^0_{n+1} = -\infty$, we can write

$$d^{AB}_j = \min\{b^R_{n+1} - a^K_{n+1}, \sum_{i=1}^{R} \min\{b^i_{n+1} - b^{i-1}_{n+1}, b^i\}\}.$$

Let $J_{i^*} = \{j \in \{1,\ldots,n\} : \ b^{i^*}_j > 0\}$. Then

$$d^{AB}_j = \begin{cases} \min\{b^R_{n+1} - a^K_{n+1}, b^{i^*}_{n+1} - b^{i^*-1}_{n+1}, b^{i^*}_j\} & \text{if } j \in J_{i^*} \\ \min\{b^R_{n+1} - a^K_{n+1}, \sum_{i=1, i\neq i^*}^{R} \min\{b^i_{n+1} - b^{i-1}_{n+1}, b^i_j\}\} & \text{if } j \notin J_{i^*}, \end{cases}$$

$$d^C_j = \begin{cases} \min\{b^{i^*}_{n+1} - a^K_{n+1}, b^{i^*}_j\} & \text{if } j \in J_{i^*} \\ \sum_{i=i^*+1}^{R} \min\{b^i_{n+1} - b^{i-1}_{n+1}, b^i_j\}\} & \text{if } j \notin J_{i^*}, \end{cases}$$

By definition of i^*, we have $b^{i^*}_{n+1} - a^K_{n+1} \leq \min\{b^{i^*}_{n+1} - b^{i^*-1}_{n+1}, b^R_{n+1} - a^K_{n+1}\}$. Clearly, $\sum_{i=i^*+1}^{R} \min\{b^i_{n+1} - b^{i-1}_{n+1}, b^i_j\} \leq \sum_{i=1,i\neq i^*}^{R} \min\{b^i_{n+1} - b^{i-1}_{n+1}, b^i_j\}$. Moreover,

$$\begin{aligned}\sum_{i=i^*+1}^{R} \min\{b^i_{n+1} - b^{i-1}_{n+1}, b^i_j\} &\leq \sum_{i=i^*+1}^{R} (b^i_{n+1} - b^{i-1}_{n+1}) \\ &= b^R_{n+1} - b^{i^*}_{n+1} \leq b^R_{n+1} - a^K_{n+1}.\end{aligned}$$

Consequently, $d^C \leq d^{AB}$, and therefore $\Delta(C) \leq \Delta(A) \circ \Delta(B)$. □

As before, we let $\Phi(A) \in \mathbb{R}^{n+1}$ be a vector obtained by an arbitrary sequence of pairings of the vectors in A.

Theorem 5.7 *If $A = \{a^1,\ldots,a^K\}$ is disjoint, then for any $\Phi(A)$, there exists $\hat{A} \subseteq A$ with $a^K \in \hat{A}$ such that*

$$\Delta(\hat{A}) \succeq \Phi(A).$$

Proof: The proof is by induction on $|A|$. The claim holds trivially for any disjoint set A such that $|A| \leq 2$. Assume that the claim holds for any disjoint set A with $|A| \leq k$.

Consider a disjoint set A such that $|A| = k+1$. Given $\Phi(A)$ obtained by an arbitrary sequence of pairings of the vectors in A, we can write

$$\Phi(A) = \Phi(A_1) \circ \Phi(A_2)$$

for some $A_1, A_2 \subset A$ such that $A_1 \cap A_2 = \emptyset$ and $A_1 \cup A_2 = A$. Note that $|A_1| \leq k$ and $|A_2| \leq k$. Thus by our induction hypothesis, there exists $\hat{A}_1 \subseteq A_1$ and $\hat{A}_2 \subseteq A_2$, such that $\Delta(\hat{A}_1) \succeq \Phi(A_1)$ and $\Delta(\hat{A}_2) \succeq \Phi(A_2)$. Then from Lemma 5.2,

$$\Phi(A) \preceq \Delta(\hat{A}_1) \circ \Delta(\hat{A}_2).$$

By Lemma 5.4, there exists a subset $\hat{A} \subseteq (\hat{A}_1 \cup \hat{A}_2) \subseteq A$ such that

$$\Phi(A) \preceq \Delta(\hat{A}_1) \circ \Delta(\hat{A}_2) \preceq \Delta(\hat{A}).$$

$\square$

As a consequence of Theorem 5.7, among all inequalities obtained by pairings of the vectors in a disjoint set A, it is sufficient to consider the inequalities corresponding to the $2^K - 1$ vectors in $C = \{\Delta(\hat{A}) : \ \hat{A} \subseteq A, \hat{A} \neq \emptyset\}$.

Even though it suffices to consider the inequalities defined by the set C, the number of such inequalities is exponential in K. Here we present a polynomial time separation algorithm for finding a most violated inequality in C if one exists. The algorithm is based on solving shortest path problems on a directed graph G with nodes $\mathcal{N} = \{0, 1, \ldots, K\}$ and arcs (i, j) for all i and $j > i$. Given a point y^*, the separation problem of determining whether there exists any violated pairing inequalities can be reduced to finding a shortest path from node 0 to node k for $1 \leq k \leq K$ where the length of arc (i, j) is given by $\sum_{r=1}^{n} \min\{a_r^j, a_{n+1}^j - a_{n+1}^i\} y_r^*$ for $i > 0$ and $\sum_{r=1}^{n} a_r^j y_r^*$ for $i = 0$. This is true because a path $P = (0, i_1, i_2, \ldots, i_k)$ in G corresponds to a matrix $\hat{A} = (a^{i_1}, a^{i_2}, \ldots, a^{i_k})$ since the length of the path is equal to the left-hand side of the inequality $\Delta(\hat{A})$. Note that by Theorem 5.6, the left-hand side of the inequality $\Delta(\hat{A})$ is $\sum_{r=1}^{n} a_r^{i_1} y_r^* + \sum_{j=2}^{k} \sum_{r=1}^{n} \min\{a_r^{i_j}, a_{n+1}^{i_j} - a_{n+1}^{i_{j-1}}\} y_r^*$, which is exactly the length of P. Therefore, there is a violated inequality with right-hand side a_{n+1}^k if and only if the length of a shortest path from 0 to k is less than a_{n+1}^k. Using Dijkstra's algorithm the separation problem can be solved in $O(K^2)$ time and we can find as many as K violated inequalities from the shortest paths from 0 to k for $k = 1, \ldots, K$.

Now we give sufficient conditions for the inequalities in C to be facet-defining for a certain class of disjoint systems. Let $A = \{a^1, \ldots, a^K\} \in \mathbb{R}^{n+1}$ be a disjoint set, and consider the mixed 0-1 set

$$X = \Big\{(y, x) \in \{0,1\}^n \times \mathbb{R}_+ : \sum_{j=1}^{n} a_j^i y_j + x \geq a_{n+1}^i,\ i = 1, \ldots, K\Big\},$$

with one continuous variable. Without loss of generality, as in the nested set case, we assume that $a_j^i \leq a_{n+1}^i$ for all $j = 1, \ldots, n$ and $i = 1, \ldots, K$. We also assume that

$$\sum_{j=1}^{n} a_j^i \geq a_{n+1}^i, \quad i = 1, \ldots, K, \tag{5.9}$$

since otherwise, we can replace x by $x + (a_{n+1}^i - \sum_{j=1}^n a_j^i)$. Consider $\hat{A} = \{a^{q_1}, \ldots, a^{q_Q}\} \subseteq A$. Let $\mathcal{Q} = \{q_1, \ldots, q_Q\}$ and, for brevity, let $q = q_1$, $Q = q_Q$. Define $\hat{\Delta} = \Delta(\hat{A})$, where the jth element $\hat{\Delta}_j$ is given by $\hat{\Delta}_j = \min\{a_{n+1}^{r(j)} - a_{n+1}^{c(r(j))}, a_j^{r(j)}\}$, with $r(j) = \{i \in \{1, \ldots, n\} : a_j^i > 0\}$ for $j = 1, \ldots, n$ and $c(i) = \operatorname{argmax}\{k \in \mathcal{Q} : k < i\}$ for all $i \in \mathcal{Q}$.

Theorem 5.8 *Given $\hat{A} \subseteq A$ and the corresponding index set $\mathcal{Q}$, the sequential pairing inequality*

$$\sum_{j=1}^{n} \hat{\Delta}_j y_j + x \geq a_{n+1}^Q \tag{5.10}$$

is facet-defining for conv(X) if

(a) $\max\{a_j^i : j \in \{1, \ldots, n\}\} \geq \max\{a_j^q : j \in \{1, \ldots, n\}\} + a_{n+1}^i - a_{n+1}^q$, *for all* $i \in \mathcal{Q}$.

(b) $\sum_{j=1}^n a_j^i \geq a_{n+1}^i - a_{n+1}^Q + a_k^i$, *for all* $k \in \{1, \ldots, n\}$ *and* $i \in \{Q+1, \ldots, K\}$.

Proof: We construct $\dim(X) = n + 1$ linearly independent vectors belong to X that satisfy (5.10) at equality.

We construct a vector corresponding to each of the $n+1$ variables. Denote $s(i) = \operatorname{argmax}\{a_j^i :$

$j \in \{1,\ldots,n\}\}$ for all $i \in \mathcal{Q}$. Let e^x be the unit vector in $\mathbb{R}^{n+1}$ corresponding to the coordinate x and e^{y_j} be the unit vector in $\mathbb{R}^{n+1}$ corresponding to the coordinate y_j for $j = 1,\ldots,n$. Let $Z(\mathcal{Q}) = \{j \in \{1,\ldots,n\} : \ \exists\, i \in \mathcal{Q}$ such that $a^i_j > 0\}$ and $\overline{Z}(\mathcal{Q}) = \{1,\ldots,n\} \setminus Z(\mathcal{Q})$. We construct the following $n+1$ vectors, denoted by $\{u^j\}_{j=0}^n$.

(i) Vector u^0 corresponds to variable x and is given by

$$u^0 = a^Q_{n+1} e^x + \sum_{i \in \overline{Z}(\mathcal{Q})} e^{y_i}.$$

(ii) For each $j \in \overline{Z}(\mathcal{Q})$, the corresponding vector u^j is given by

$$u^j = u^0 - e^{y_j}.$$

(iii) For each $j \in Z(\mathcal{Q})$, the corresponding vector u^j is given by

$$u^j = (a^{r(j)}_{n+1} - \hat{\Delta}_j) e^x + \sum_{i \in \overline{Z}(\mathcal{Q})} e^{y_i} + e^{y_j} + \sum_{i \in \mathcal{Q},\ i > r(j)} e^{y_{s(i)}}.$$

Feasibility: We need to show that $\{u^j\}_{j=0}^n$ satisfies

$$a^k u \geq a^k_{n+1} \qquad k = 1,\ldots,K, \tag{5.11}$$

$$u_0 \geq 0,\ u_j \in \{0,1\} \qquad j = 1,\ldots,n, \tag{5.12}$$

where $a^k_0 = 1$ for all $k = 1,\ldots,K$.

(i) The feasibility of u^0 is based on (5.9).

(ii) The feasibility of u^j for each $j \in \overline{Z}(\mathcal{Q})$ is based on condition (b).

(iii) For a given $j \in Z(\mathcal{Q})$, the vector u^j satisfies (5.12) since $\hat{\Delta}_j \leq a^{r(j)}_j \leq a^{r(j)}_{n+1}$. The left-hand of (5.11) corresponding to $i \in \{1,\ldots,K\} \setminus \mathcal{Q}$ is

$$a^{r(j)}_{n+1} - \hat{\Delta}_j + \textstyle\sum_{j=1}^n a^i_j u^j_j \ \ \geq \ \ \textstyle\sum_{j=1}^n a^i_j \geq a^i_{n+1},$$

where the first inequality follows from $\hat{\Delta}_j \leq a^{r(j)}_j \leq a^{r(j)}_{n+1}$ and the second inequality follows from (5.9).

The left-hand side of (5.11) corresponding to $i \in \mathcal{Q}$ and $i = r(j)$ is

$$a_{n+1}^{r(j)} - \hat{\Delta}_j + \sum_{j=1}^{n} a_j^i u_j^j = a_{n+1}^{r(j)} - \hat{\Delta}_j + a_j^{r(j)} \geq a_{n+1}^{r(j)} = a_{n+1}^i,$$

where the inequality follows from the definition of $\hat{\Delta}_j$.

The left-hand side of (5.11) corresponding to $i \in \mathcal{Q}$ and $i < r(j)$ is

$$\begin{aligned} a_{n+1}^{r(j)} - \hat{\Delta}_j + \textstyle\sum_{j=1}^{n} a_j^i u_j^j &\geq a_{n+1}^{r(j)} - \hat{\Delta}_j \\ &\geq a_{n+1}^{r(j)} - (a_{n+1}^{r(j)} - a_{n+1}^{c(r(j))}) \\ &= a_{n+1}^{c(r(j))} \geq a_{n+1}^i, \end{aligned}$$

where the second inequality follows from the definition of $\hat{\Delta}_j$.

The left-hand side of (5.11) corresponding to $i \in \mathcal{Q}, i > r(j)$ and $r(j) = q$ is

$$\begin{aligned} a_{n+1}^{r(j)} - \hat{\Delta}_j + \textstyle\sum_{j=1}^{n} a_j^i u_j^j &= a_{n+1}^{r(j)} - \hat{\Delta}_j + a_{s(i)}^i \\ &\geq a_{n+1}^q - a_j^q + a_{s(i)}^i \\ &\geq a_{n+1}^q - a_j^q + \max\{a_j^q : j \in \{1, \ldots, n\}\} + a_{n+1}^i - a_{n+1}^q \\ &\geq a_{n+1}^i, \end{aligned}$$

where the first inequality follows from $r(j) = q$, $\hat{\Delta}_j \leq a_j^q$ and the second inequality follows from condition (a).

The left-hand side of (5.11) corresponding to $i \in \mathcal{Q}, i > r(j)$ and $r(j) \neq q$ is

$$\begin{aligned} a_{n+1}^{r(j)} - \hat{\Delta}_j + \textstyle\sum_{j=1}^{n} a_j^i u_j^j &= a_{n+1}^{r(j)} - \hat{\Delta}_j + a_{s(i)}^i \\ &\geq a_{n+1}^{r(j)} - (a_{n+1}^{r(j)} - a_{n+1}^{c(r(j))}) + a_{s(i)}^i \\ &= a_{n+1}^{c(r(j))} + a_{s(i)}^i \\ &\geq a_{n+1}^{c(r(j))} + \max\{a_j^q : j \in \{1, \ldots, n\}\} + a_{n+1}^i - a_{n+1}^q \\ &\geq a_{n+1}^i, \end{aligned}$$

where the first inequality follows from the definition of $\hat{\Delta}_j$, the second inequality follows from condition (a) and the third inequality follows from the fact that $a_{n+1}^{c(r(j))} \geq a_{n+1}^q$.

Tightness:

(i, ii) It is easily verified that u^0 and u^j for each $j \in \overline{Z}(\mathcal{Q})$ satisfy (5.10) as an equality.

(iii) For a given $j \in Z(\mathcal{Q})$, the left-hand side of (5.10) corresponding to u^j is

$$\begin{aligned} u_0^j + \textstyle\sum_{i=1}^n \hat{\Delta}_i u_i^j &= (a_{n+1}^{r(j)} - \hat{\Delta}_j) + \hat{\Delta}_j + \textstyle\sum_{i \in \mathcal{Q}, i > r(j)} \hat{\Delta}_{s(i)} \\ &= a_{n+1}^{r(j)} + \textstyle\sum_{i \in \mathcal{Q}, i > r(j)} (a_{n+1}^i - a_{n+1}^{c(i)}) = a_{n+1}^{Q}, \end{aligned}$$

where the second equality follows from

$$\hat{\Delta}_{s(i)} = \min\{a_{n+1}^i - a_{n+1}^{c(i)}, a_{s(i)}^i\}, \text{ and}$$

$$a_{s(i)}^i = \max\{a_j^i : j \in \{1, \ldots, n\}\} \geq a_{n+1}^i - a_{n+1}^q \geq a_{n+1}^i - a_{n+1}^{c(i)},$$

which follows from (a).

Linear independence: To verify the linear independence of the $n+1$ vectors $\{u^j\}_{j=0}^n$, we can obtain the following $n+1$ vectors from $\{u^j\}_{j=0}^n$ as follows:

$$\begin{aligned} e^{y_j} &= u^0 - u^j, \quad \text{for each } \ j \in \overline{Z}(\mathcal{Q}). \\ e^x &= u^0 - \sum_{i \in \overline{Z}(\mathcal{Q})} e^{y_i}. \\ v^j &= u^j - (a_{n+1}^{r(j)} - \hat{\Delta}_j) e^x - \sum_{i \in \overline{Z}(\mathcal{Q})} e^{y_i} \\ &= e^{y_j} + \sum_{i \in \mathcal{Q}, i > r(j)} e^{y_{s(i)}}, \text{for each } j \in Z(\mathcal{Q}). \end{aligned}$$

By sorting v^j according to the decreasing sequence of $r(j)$, it can be verified that v^j for each $j \in Z(\mathcal{Q})$ forms a lower triangular. Therefore, these vectors are linearly independent, which implies that original vectors are linearly independent. $\square$

5.5 Applications

Dynamic knapsack sets: Consider the set X given by (5.4) with $a \in \mathbb{R}_+^n$ and $d \in \mathbb{R}_+^n$. Let $d_{ij} = \sum_{k=i}^j d_k$, Loparic et al. [67] proved that the inequality

$$x + \sum_{j=1}^{i} \min\{a_j, d_{ji}\} y_j \geq d_{1i} \tag{5.13}$$

is valid for $\text{conv}(X)$ for $i = 1, \ldots, n$, and facet-defining when $i = n$. Dynamic knapsack sets are nested. Applying the pairing sequence Δ to the inequalities (5.4) gives the inequalities (5.13). $i = n$ corresponds to $i = K$ in Theorem 5.5, and the inequality corresponding to $i = n$ satisfies the facet-defining conditions (a) and (b) in Theorem 5.5. We also notice that conditions (a) and (b) in Theorem 5.5 provide more facet-defining inequalities for dynamic knapsack sets.

Mixed vertex packing: The mixed vertex packing problem (MVP) is a generalization of the vertex packing problem having both binary and bounded continuous variables. Let N denote the index set of binary variables, M denote the index set of continuous variables and $N(k) = \{i \in N : (k, i) \in E \cup F\}$, where $E \subseteq \{(i, j) : i, j \in N\}$ is defined as the *binary edge set* and $F \subseteq \{(i, k) : i \in N, k \in M\}$ is defined as the *mixed edge set*. The feasible solution set of MVP is

$$
\begin{aligned}
X_{\text{MVP}} = \quad & \Big\{(y, x) \in \{0, 1\}^n \times \mathbb{R}^m : \\
& \qquad y_i + y_j \le 1, \qquad\qquad (i, j) \in E && (5.14)\\
& \qquad a_{ik} y_i + x_k \le u_k, \qquad (i, k) \in F && (5.15)\\
& \qquad 0 \le x_k \le u_k, \qquad\qquad k \in M \Big\}.
\end{aligned}
$$

For each $k \in M$, let $T = \{i_1, i_2, \ldots, i_t\} \subset N(k)$ such that $a_{i_{j-1}k} < a_{i_j k}$ for $j = 2, 3, \ldots, t$. Atamtürk et al. [9] showed that the *star* inequality

$$\sum_{i \in T} \bar{a}_{ik} y_i + x_k \le u_k, \tag{5.16}$$

where $\bar{a}_{i_1 k} = a_{i_1 k}$ and $\bar{a}_{i_j k} = a_{i_j k} - a_{i_{j-1} k}$ for $j = 2, \ldots, t$, is valid for X_{MVP}. Note that the *mixed edge set* inequalities form a disjoint set with respect to the binary variables.

We now show that the pairing scheme can generate all of the *star* inequalities. By complementing the binary variables for the *mixed edge set* inequalities (5.15) corresponding to edge $(i, k) \in F, i \in T$, we have

$$a_{ik} \bar{y}_i - x_k \ge a_{ik} - u_k, \;\; (i, k) \in F, \;\; i \in T \tag{5.17}$$

where $\bar{y}_i = 1 - y_i$. Applying the pairing sequence Δ to (5.17), we obtain

$$\sum_{i \in T} \bar{a}_{ik} \bar{y}_i - x_k \geq a_{i_t k} - u_k$$

with $\bar{a}_{i_1 k} = a_{i_1 k}$ and $\bar{a}_{i_j k} = a_{i_j k} - a_{i_{j-1} k}$ for $j = 2, \ldots, t$. That is,

$$\sum_{i \in T} \bar{a}_{ik} (1 - y_i) - x_k \geq a_{i_t k} - u_k,$$

which is exactly the *star* inequality (5.16). It is also shown in [9] that the *star* inequality is facet-defining for $\text{conv}(X_{\text{MVP}})$ if $a_{i_t k} = \max_{j \in N(k)} a_{jk}$ and $N(i) = \emptyset$ for all $i \in T$. If $a_{i_t k} = \max_{j \in N(k)} a_{jk}$, then facet-defining conditions (a) and (b) in Theorem 5.8 are also satisfied by the equivalent formulation (5.17). The condition (b) is trivially true since $a_{i_t k} = \max_{j \in N(k)} a_{jk}$ corresponds to $Q = K$ for the disjoint case in Theorem 5.8 and condition (a) is also satisfied since the inequalities in condition (a) always hold at equality.

Deterministic lot-sizing: As mentioned in Chapter 1, the deterministic uncapacitated lot-sizing problem is to minimize total production and inventory holding cost while satisfying demand over a finite discrete-time planning horizon. Let x_i be the production in period i, $y_i \in \{0, 1\}$ indicate if there is a production set-up in period i, d_i be the demand in period $i \in \{1, \ldots, n\}$, and $d_{st} = \sum_{i=s}^{t} d_i$. The feasible solution set of the lot-sizing problem is

$$X_{\text{LS}} = \Big\{ (y, x) \in \{0, 1\}^n \times \mathbb{R}^n_+ : \ \sum_{j=1}^{i} x_j \geq d_{1i}, \ 0 \leq x_i \leq d_{in} y_i, \quad i = 1, \ldots, n \Big\}.$$

Barany et al. [12] described the convex hull of X_{LS} by introducing the (ℓ, S) inequalities

$$\sum_{i \in S} x_i + \sum_{i \in L \setminus S} d_{i\ell} y_i \geq d_{1\ell} \tag{5.18}$$

for $1 \leq \ell \leq n$, $L = \{1, \ldots, \ell\}$ and $S \subseteq L$.

We now show that the pairing scheme can generate all of the (ℓ, S) inequalities. For given ℓ and S, we use the constraints $\sum_{j \leq k} x_j \geq d_{1k}$ for each $k \leq \ell$ and $x_j \leq d_{jn} y_j$ for each $j \in \{1, \ldots, n\}$ to obtain the inequalities

$$\sum_{j \in S_k} x_j + \sum_{j \in L_k \setminus S_k} d_{jn} y_j \geq d_{1k} \text{ for each } k \leq \ell, \tag{5.19}$$

where $L_k = \{1, 2, \ldots, k\}$ and $S_k = S \cap L_k$. The family of inequalities (5.19) is nested (note here $1 \in S_k$ for each $k \leq \ell$). By Theorem 5.2, applying sequential pairing to (5.19) provides the (ℓ, S) inequality in (5.18) since we have $\Delta_j^\ell = \min\{d_{1\ell} - d_{11}, \ldots, d_{1\ell} - d_{1(j-1)}, d_{jn} + d_{1\ell} - d_{1j}, \ldots, d_{jn}\} = d_{1\ell} - d_{1(j-1)} = d_{j\ell}$ corresponding to each $j \in L \setminus S$.

Stochastic lot-sizing: As mentioned in Chapter 4, the stochastic uncapacitated lot-sizing problem is the stochastic programming extension of the deterministic formulation. Instead of deterministic cost and demand information for each time period, the problem parameters are random and evolve as discrete time stochastic processes with a finite probability space. A scenario tree is used to model this information where each node i in stage t of the tree represents a possible state of the system. For each node i, let $\mathcal{T}(i) = (\mathcal{V}(i), \mathcal{E}(i))$ be the subtree containing all descendants of node i, $\mathcal{L}(i)$ be the leaf nodes of the subtree $\mathcal{T}(i)$, $\mathcal{P}(i, j)$ be the set of nodes on the path from node i to node j and $d_{ij} = \sum_{k \in \mathcal{P}(i,j)} d_k$, where d_i represents the demand in period $t(i)$ for node i. For brevity, let $\mathcal{T} = \mathcal{T}(0), \mathcal{V} = \mathcal{V}(0), \mathcal{L} = \mathcal{L}(0)$ and $\mathcal{P}(i) = \mathcal{P}(0, i)$.

Let x_i be the production and y_i be the indicator variable for a production set-up in period $t(i)$ corresponding to the state defined by node i. The feasible solution set of the stochastic lot-sizing problem as shown in Chapter 4 is

$$X_{\text{SLS}} = \Big\{(y, x) \in \{0,1\}^n \times \mathbb{R}_+^n : \sum_{j \in \mathcal{P}(i)} x_j \geq d_{0i},\ 0 \leq x_i \leq M_i y_i, \quad i \in \mathcal{V}\Big\},$$

where $M_i = \max_{j \in \mathcal{L}(i)} d_{ij}$ is an upper bound on x_i.

We developed a family of valid inequalities for X_{SLS} called the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequalities in Chapter 4. Here we use the same notation except that $\delta_{\mathcal{Q}}(i) = \min\Big\{\overline{D}_{\mathcal{Q}}(i) - \widetilde{D}_{\mathcal{Q}}(i), M_{\mathcal{Q}}(i)\Big\}$ for convenience.

We can use sequential pairing to generate all $(\mathcal{Q}, S_{\mathcal{Q}})$ inequalities. Given a $(\mathcal{Q}, S_{\mathcal{Q}})$ tuple, first, we can use sequential pairing, as in the deterministic lot-sizing case, to generate (ℓ, S) inequalities corresponding to $\mathcal{P}(i)$ for each $i \in \mathcal{Q}$ as

$$\sum_{j \in \mathcal{P}(i) \cap S_{\mathcal{Q}}} x_j + \sum_{j \in \mathcal{P}(i) \cap \overline{S}_{\mathcal{Q}}} d_{ji} y_j \geq d_{0i}. \tag{5.20}$$

Then, we use sequential pairing of the inequalities (5.20) for $i = 1$ to Q to obtain

$$\sum_{i \in S_{\mathcal{Q}}} x_i + \sum_{i \in \overline{S}_{\mathcal{Q}}} \delta_{\mathcal{Q}}(i) y_i \geq d_{0Q} = M_{\mathcal{Q}}(0). \tag{5.21}$$

To see that sequential pairing leads to the correct coefficients in (5.21), note that this claim is clearly true for $|\mathcal{Q}| = 1$ since this case is exactly that of an (ℓ, S) inequality for the deterministic lot-sizing problem. Assuming that the claim is true for $|\mathcal{Q}| = k$, we have

$$\sum_{i \in S_{\mathcal{Q}} \cap \mathcal{V}_{\mathcal{Q}_k}} x_i + \sum_{i \in \overline{S}_{\mathcal{Q}} \cap \mathcal{V}_{\mathcal{Q}_k}} \delta_{\mathcal{Q}_k}(i) y_i \geq d_{0k},$$

where $\mathcal{Q}_k = \{1, 2, \ldots, k\}$. By pairing the above inequality with the (ℓ, S) inequality

$$\sum_{j \in \mathcal{P}(k+1) \cap S_{\mathcal{Q}}} x_j + \sum_{j \in \mathcal{P}(k+1) \cap \overline{S}_{\mathcal{Q}}} d_{j(k+1)} y_j \geq d_{0(k+1)}$$

corresponding to $i = k + 1$, the resulting coefficients corresponding to each $j \in \overline{S}_{\mathcal{Q}}$ are as follows.

(i) The coefficient corresponding to each $i \in \mathcal{V}_{\mathcal{Q}_k} \setminus \mathcal{P}(k + 1)$ remains unchanged and $\delta_{\mathcal{Q}_k}(i) = \delta_{\mathcal{Q}_{k+1}}(i)$.

(ii) The coefficient corresponding to each $i \in \mathcal{P}(k + 1) \setminus \mathcal{V}_{\mathcal{Q}_k}$ is equal to $\min\{d_{0(k+1)} - d_{0k}, d_{i(k+1)}\}$, which is $\delta_{\mathcal{Q}_{k+1}}(i)$.

(iii) The coefficient corresponding to each $i \in \mathcal{P}(k+1) \cap \mathcal{V}_{\mathcal{Q}_k}$ is equal to $\delta_{\mathcal{Q}_k}(i) + d_{0(k+1)} - d_{0k} = \delta_{\mathcal{Q}_{k+1}}(i)$ since $M_{\mathcal{Q}_{k+1}}(i) = M_{\mathcal{Q}_k}(i) + d_{0(k+1)} - d_{0k}$, $\widetilde{D}_{\mathcal{Q}_{k+1}}(i) = \widetilde{D}_{\mathcal{Q}_k}(i)$ and $\overline{D}_{\mathcal{Q}_{k+1}}(i) = \overline{D}_{\mathcal{Q}_k}(i) + d_{0(k+1)} - d_{0k}$.

Thus we have the correct coefficients in (5.21).

5.6 Computational Experiments

In this section we provide some numerical results to demonstrate the computational effectiveness of the pairing scheme on randomly generated instances of mixed-integer programs with nested and disjoint sets of constraints. All computations have been carried out on a Linux workstation with dual 2.4 GHz Intel Xeon processors and 2 GB RAM using CPLEX 8.1.

For the nested case, we generated random instances of the model

$$\begin{aligned}
\min \quad & \sum_{j=1}^{mn} c_j y_j + \sum_{k=1}^{p} h_k x_k \\
& \sum_{j=1}^{in} a_j^i y_j + \sum_{k=1}^{p} g_k^i x_k \geq b_i \quad && i = 1, \ldots, m \\
& y_j \in \{0, 1\} && j = 1, \ldots, mn \\
& x_k \geq 0 && k = 1, \ldots, p.
\end{aligned}$$

This model has n additional binary variables in each successive row, with a total of mn binary variables and p continuous variables. The constraint coefficients and the right-hand sides were generated such that these form a nested system and were uniformly distributed within the interval $[50, 75]$ and $[50, 100]$, respectively. The objective function coefficients were uniformly distributed within the interval $[10, 100]$. In Table 8, we present computational results for $p \in \{1, 2, 3\}$, $n \in \{1, 2, 3\}$ and $m \in \{10, 20, 40\}$. For each combination of m, n, and p, we tested five instances and report the average objective function value in the column labelled "OptVal." The row labelled "LP" provides the average optimal objective value of the linear programming relaxation without any cuts; the row labelled "LP+CUTS" (LPC) provides the average optimal objective value after adding all inequalities obtained through pairing as cuts, which can be done since the total number of cuts is small and equal to the number of rows; and the row labelled "IP" provides the optimal value of the corresponding integer programming problem. The column labelled "Gap" provides the percentage LP relaxation gap, computed as (IP-LP)/LP$\times$100% and (IP-LPC)/LPC$\times$100%. We observe that the cuts yield significant improvements. In 13 of the 27 cases, the gap is reduced to 0% from over 10%. In all but three of the cases, the gap is reduced by more than half.

For the disjoint case, we generated random instances of the model

$$\begin{aligned}
\min \quad & \sum_{i=1}^{m} \sum_{j=1}^{n} c_j^i y_j^i + \sum_{k=1}^{p} h_k x_k \\
& \sum_{j=1}^{n} a_j^i y_j^i + \sum_{k=1}^{p} g_k x_k \geq b_i \quad && i = 1, \ldots, m \\
& y_j^i \in \{0, 1\} && j = 1, \ldots, n, i = 1, \ldots, m \\
& x_k \geq 0 && k = 1, \ldots, p.
\end{aligned}$$

Table 8: Computational Results for the Nested Case

			$m = 10$		$m = 20$		$m = 40$	
p	n		OptVal	Gap	OptVal	Gap	OptVal	Gap
		LP	100.59	19.58%	51.10	13.97%	24.43	22.18%
1	1	LP+CUTS	114.41	8.53%	59.40	0.00%	31.18	0.67%
		IP	125.08		59.40		31.39	
		LP	65.88	23.63%	48.87	13.25%	21.21	15.99%
1	2	LP+CUTS	77.80	9.82%	56.33	0.00%	25.02	0.89%
		IP	86.27		56.33		25.25	
		LP	39.71	19.83%	48.19	14.50%	21.43	14.56%
1	3	LP+CUTS	43.27	12.64%	56.36	0.00%	24.95	0.51%
		IP	49.53		56.36		25.08	
		LP	23.00	4.61%	31.47	10.38%	65.86	13.05%
2	1	LP+CUTS	24.11	0.00%	35.12	0.00%	75.75	0.00%
		IP	24.11		35.12		75.75	
		LP	22.62	9.50%	31.45	11.65%	58.21	15.77%
2	2	LP+CUTS	24.99	0.00%	35.60	0.00%	66.96	3.11%
		IP	24.99		35.60		69.11	
		LP	22.02	7.92%	31.42	13.90%	56.89	15.39%
2	3	LP+CUTS	23.92	0.00%	36.49	0.00%	63.95	4.87%
		IP	23.92		36.49		67.23	
		LP	20.28	19.45%	21.99	30.57%	69.13	14.96%
3	1	LP+CUTS	24.18	3.95%	28.03	11.52%	81.29	0.00%
		IP	25.18		31.68		81.29	
		LP	17.05	28.54%	20.39	27.60%	64.66	13.35%
3	2	LP+CUTS	20.47	14.20%	22.81	18.99%	74.62	0.00%
		IP	23.86		28.16		74.62	
		LP	18.99	25.28%	20.06	29.40%	64.13	11.93%
3	3	LP+CUTS	22.52	11.41%	22.74	19.96%	72.82	0.00%
		IP	25.42		28.41		72.82	

Each row of this model has n independent binary variables giving rise to a disjoint system involving a total of mn binary variables. A total of p continuous variables couple the binary variables together. The constraint coefficients and the right-hand sides were generated uniformly within the interval $[40, 120]$ and $[100, 125]$ respectively. The objective function coefficients were uniformly distributed within the interval $[10, 100]$ for the continuous variables and within the interval $[10/m, 100/m]$ for the binary variables. In Table 9, we present computational results corresponding to $p \in \{1, 2, 3\}$, $n \in \{1, 2, 3\}$ and $m \in \{10, 20, 40\}$. As before, we report averages over five random instances for each combination of m, n and p. In this case, we use the shortest path separation routine described in Section 5.4 to add only violated cuts. The average number of cuts added is reported in the row labelled "#

CUTS." Once again, we observe that the cuts yield significant improvements. In 6 of the 27 cases, the gap is reduced to 0%. In 19 of the 27 cases, the gap is reduced by more than half. The number of cuts ranges from 30, on average for 10 rows, to 491, on average for 40 rows.

Table 9: Computational Results for the Disjoint Case

			$m = 10$		$m = 20$		$m = 40$	
p	n		OptVal	Gap	OptVal	Gap	OptVal	Gap
		LP	1082.24	11.62%	703.83	9.03%	874.35	12.57%
1	1	LP+CUTS	1099.88	10.17%	729.58	5.71%	928.66	7.14%
		IP	1224.47		773.73		1000.08	
		# CUTS	45		163		925	
		LP	585.63	38.48%	538.49	25.20%	702.46	29.22%
1	2	LP+CUTS	793.18	16.68%	655.58	8.93%	946.30	4.65%
		IP	952.00		719.89		992.45	
		# CUTS	32		162		1317	
1	3	LP	410.88	26.45%	446.83	23.08%	559.81	24.67%
		LP+CUTS	480.88	13.92%	452.76	22.05%	619.81	16.59%
		IP	558.66		580.87		743.13	
		# CUTS	10		27		124	
		LP	685.99	8.36%	388.66	5.96%	497.25	7.69%
2	1	LP+CUTS	693.98	7.29%	400.19	3.17%	522.35	3.03%
		IP	748.58		413.29		538.67	
		# CUTS	34		74		341	
		LP	507.76	33.72%	356.65	14.23%	464.42	13.56%
2	2	LP+CUTS	710.69	7.22%	415.34	0.12%	529.88	1.38%
		IP	766.03		415.83		537.30	
		# CUTS	39		119		530	
		LP	400.18	20.50%	339.61	12.71%	437.98	13.60%
2	3	LP+CUTS	448.58	10.89%	357.13	8.21%	470.21	7.25%
		IP	503.38		389.06		506.95	
		# CUTS	20		63		347	
		LP	533.31	2.84%	285.17	2.09%	387.16	4.80%
3	1	LP+CUTS	542.35	1.19%	291.25	0.00%	406.67	0.00%
		IP	548.88		291.25		406.67	
		# CUTS	25		34		173	
		LP	433.80	21.02%	280	3.86%	375.75	6.84%
3	2	LP+CUTS	540.33	1.63%	291.23	0.00%	403.35	0.00%
		IP	549.27		291.23		403.35	
		# CUTS	45		59		289	
		LP	420.45	14.19%	279.87	4.38%	360.44	9.80%
3	3	LP+CUTS	459.61	6.19%	292.69	0.00%	399.61	0.00%
		IP	489.95		292.69		399.61	
		# CUTS	24		67		375	

5.7 *Conclusions*

In this chapter, we have developed a new and simple way of pairwise combining linear inequalities for MIPs to obtain new linear inequalities. These new inequalities can be useful in tightening the LP relaxation for general MIPs. The order in which the inequalities are combined can have a significant impact on the results. For some structured systems, we provided combination orders that are optimal in the sense that no other combination order can dominate the set of inequalities given by the optimal order. These structures arise in multi-period MIPs. We discussed applications of these structures to deterministic and stochastic lot-sizing problems. One of our goals is to apply the procedure to general multi-period stochastic MIPs. To do this we need to generalize the structures considered in this paper to scenario trees. This will be discussed in the next chapter.

CHAPTER 6

COMBINING $0-1$ INEQUALITIES: PATH TO TREE

In this chapter, we generalize the idea of the pairing scheme introduced in Chapter 5 to a fundamental stochastic integer programming problem, the stochastic dynamic knapsack problem, by extending the dynamic knapsack formulation provided by Loparic, Marchand and Wolsey [67] to a stochastic setting. Using a scenario tree for the uncertain parameters, this structure is fundamental to a variety of general stochastic integer programs. Thus, the results of this chapter can be applied to multi-period stochastic integer programs with a finite number of scenarios.

From the definition of the stochastic scenario tree in Section 1.1.2, the nested case described in Section 5.3 is a special case of the stochastic scenario tree corresponding to a path while the disjoint case described in Section 5.4 is another special case corresponding to a branch node. In Section 6.1, we use the idea of our pairing scheme to provide a closed form expression for a new class of valid inequalities. We present facet-defining condition for these new inequalities and give sufficient conditions such that the new family of inequalities provide the convex hull of feasible integer solutions. We analyze the complexity of separating this family of inequalities for a special case and develop a heuristic separation approach for the general case. We also give a description of the non-dominated inequalities for the tree structure generated by the pairing scheme.

In Section 6.2, we study the application of this sequential pairing scheme to stochastic lot-sizing problems. Finally, in Section 6.3, we report some preliminary computational results for the multi-item stochastic capacitated lot-sizing problem.

6.1 Stochastic Dynamic Knapsack Problem

In this section, we study a fundamental stochastic integer programming structure, called the *stochastic dynamic knapsack problem.* It extends the deterministic dynamic knapsack problem introduced by Loparic *et al.* [67] to the stochastic setting. In [67], a dynamic knapsack set is defined as

$$X_{\text{DK}} = \{(x, y) \in \mathbb{R}^1_+ \times \mathcal{B}^n : x + \sum_{j:j \leq t} a_j y_j \geq b_t \text{ for each } t \in \mathcal{N}\} \tag{6.1}$$

where $\mathcal{N} = \{1, \ldots, n\}, a \in \mathbb{R}^n_+$ and $b \in \mathbb{R}^n$. This set generalizes the knapsack set with a single continuous variable studied in [73]. In this chapter, we extend this deterministic formulation to a stochastic dynamic knapsack set with a single continuous variable defined by

$$X_{\text{SDK}} = \{(x, y) \in \mathbb{R}^1_+ \times \mathcal{B}^n : x + \sum_{j \in \mathcal{P}(i)} a_j y_j \geq b_i \text{ for each } i \in \mathcal{V} \text{ and } x \leq b_{\mathcal{V}}\} \tag{6.2}$$

where $|\mathcal{V}| = n, a \in \mathbb{R}^n_+$, $b \in \mathbb{R}^n$ and $b_{\mathcal{V}} = \max\{b_i, i \in \mathcal{V}\}$. In the following sections, without loss of generality, we assume $b_j \leq b_i$ if $j \in \mathcal{P}(i)$.

Remark 6.1 *When $|\mathcal{L}| = 1$, the above formulation is equivalent to the dynamic knapsack problem. It is also the same as the nested case studied in Chapter 5.*

Remark 6.2 *When $T = 2$, the above formulation is the same as the disjoint case studied in Chapter 5.*

6.1.1 The New Inequalities

Theorem 6.1 *Given a set $R = \{1, 2, \ldots, Q\} \subseteq \mathcal{V}$ indexed such that $b_1 \leq b_2 \leq \ldots \leq b_Q$, the inequality*

$$x + \sum_{j \in \mathcal{V}_R} \phi_R(j) y_j \geq b_Q, \tag{6.3}$$

where $\mathcal{V}_R = \cup_{i \in R} \mathcal{P}(i)$ and $\phi_R(j) = \min\{a_j, \sum_{i \in R(j)}(b_i - b_{i-1})\}$ with $R(j) = R \cap \mathcal{V}(j)$ and $b_0 = 0$, is valid for X_{SDK}.

Proof: The proof is induction over $k \in \{1, 2, \ldots, Q\}$ such that the inequality

$$x + \sum_{j \in \mathcal{V}_{R_k}} \phi_{R_k}(j) y_j \geq b_k \tag{6.4}$$

where $R_k = \{1, 2, \ldots, k\}$ is valid for X_{SDK}.

The base case ($k = 1$): According to the definition of X_{SDK}, the inequality

$$x + \sum_{j \in \mathcal{P}(1)} a_j y_j \geq b_1 \tag{6.5}$$

is valid for X_{SDK} in the original formulation in (6.2). After coefficient tightening, the inequality

$$x + \sum_{j \in \mathcal{P}(1)} \min\{b_1, a_j\} y_j \geq b_1 \tag{6.6}$$

is valid. Note that when $R = \{1\}$, we have $\mathcal{V}_R = \cup_{i \in R} \mathcal{P}(i) = \mathcal{P}(1)$ and $R(j) = 1$ for each $j \in \mathcal{P}(1)$. Then $\phi_R(j) = \min\{a_j, b_1 - b_0\}$ for each $j \in \mathcal{V}_R = \mathcal{P}(1)$. Thus, (6.6) is the same as (6.3) with $b_0 = 0$. Therefore, (6.3) is valid for X_{SDK} for $R = \{1\}$.

The inductive step ($k = K$): Assume the inequality

$$x + \sum_{j \in \mathcal{V}_{R_K}} \phi_{R_K}(j) y_j \geq b_K. \tag{6.7}$$

is valid for X_{SDK}.

We show that the inequality

$$x + \sum_{j \in \mathcal{V}_{R_{K+1}}} \phi_{R_K+1}(j) y_j \geq b_{K+1}. \tag{6.8}$$

is also valid for X_{SDK}.

Note that

$$x + \sum_{j \in \mathcal{P}(K+1)} a_j y_j \geq b_{K+1} \tag{6.9}$$

is valid for X_{SDK} in the original formulation in (6.2). Now we pair (6.7) and (6.9) to prove that (6.8) is valid for X_{SDK} based on Theorem 5.1. We consider three cases:

Case (a): For each $j \in \mathcal{P}(K+1) \setminus \mathcal{V}_{R_K}$, we have

$$\begin{aligned} \phi_{R_{K+1}}(j) &= \min\{b_{K+1} - b_K, \max(0, a_j)\} \\ &= \min\{b_{K+1} - b_K, a_j\} \\ &= \min\{a_j, \sum_{i \in R_{K+1}(j)} (b_i - b_{i-1})\} \end{aligned}$$

where the last equation follows from the fact that $R_{K+1}(j) = \{K+1\}$ for $j \in \mathcal{P}(K+1) \setminus \mathcal{V}_{R_K}$.

Case (b): For each $j \in \mathcal{V}_{R_K} \setminus \mathcal{P}(K+1)$, we have

$$\begin{aligned}
\phi_{R_{K+1}}(j) &= \min\{\phi_{R_K}(j) + b_{K+1} - b_K, \max(\phi_{R_K}(j), 0)\} \\
&= \min\{\phi_{R_K}(j) + b_{K+1} - b_K, \phi_{R_K}(j)\} \\
&= \phi_{R_K}(j) \\
&= \min\{a_j, \sum_{i \in R_K(j)} (b_i - b_{i-1})\} \\
&= \min\{a_j, \sum_{i \in R_{K+1}(j)} (b_i - b_{i-1})\}
\end{aligned}$$

where the first equation follows from $\phi_{R_K}(j) \geq 0$ and the last equation follows from $R_{K+1}(j) = R_K(j)$ for each $j \in \mathcal{V}_{R_K} \setminus \mathcal{P}(K+1)$.

Case (c): For each $j \in \mathcal{V}_{R_K} \cap \mathcal{P}(K+1)$, we have

$$\phi_{R_{K+1}}(j) = \min\{\phi_{R_K}(j) + b_{K+1} - b_K, \max(\phi_{R_K}(j), a_j)\} \tag{6.10}$$

where $\phi_{R_K}(j) = \min\{a_j, \sum_{i \in R_K(j)}(b_i - b_{i-1})\}$.

(1) If $a_j \leq \sum_{i \in R_K(j)}(b_i - b_{i-1})$, which is less or equal to $\sum_{i \in R_{K+1}(j)}(b_i - b_{i-1})$, then $\phi_{R_K}(j) = a_j$ and from (6.10), we have

$$\begin{aligned}
\phi_{R_{K+1}}(j) &= \min\{\phi_{R_K}(j) + b_{K+1} - b_K, a_j\} \\
&= \min\{a_j + b_{K+1} - b_K, a_j\} \\
&= a_j \qquad (6.11)
\end{aligned}$$

where the second equation follows from the fact that $b_{K+1} - b_K \geq 0$.

(2) If $a_j \geq \sum_{i \in R_K(j)}(b_i - b_{i-1})$, then $\phi_{R_K}(j) = \sum_{i \in R_K(j)}(b_i - b_{i-1})$ and from (6.10), we have

$$\begin{aligned}
\phi_{R_{K+1}}(j) &= \min\{\sum_{i \in R_K(j)} (b_i - b_{i-1}) + b_{K+1} - b_K, a_j\} \\
&= \min\{\sum_{i \in R_{K+1}(j)} (b_i - b_{i-1}), a_j\} \qquad (6.12)
\end{aligned}$$

where the second equation follows from the fact that $R_{K+1}(j) = R_K(j) \cup \{K+1\}$ for each $j \in \mathcal{V}_{R_K} \cap \mathcal{P}(K+1)$.

Combining (6.11) and (6.12), we have

$$\phi_{R_{K+1}}(j) = \min\{a_j, \sum_{i \in R_{K+1}(j)} (b_i - b_{i-1})\}.$$

Note that the right hand side value for the new inequality is b_{K+1} and therefore (6.8) is also valid for X_{SDK} for any set $R_{K+1} \subseteq \mathcal{V}$. □

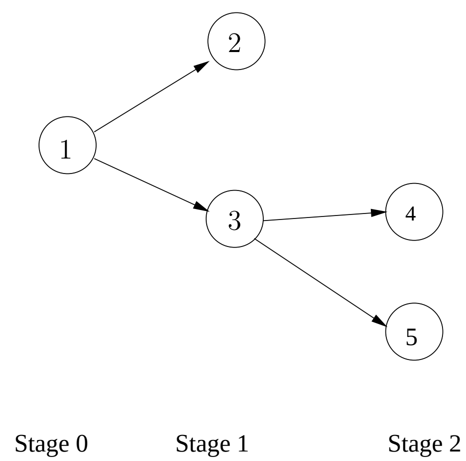

Figure 5: General scenario tree example

Example: Consider an instance of (SDK) with 5 nodes as shown in Figure 5. We set the problem parameters

$$a_1 = 40, a_2 = 15, a_3 = 20, a_4 = 20, a_5 = 40$$

and

$$b_1 = 5, b_2 = 15, b_3 = 17, b_4 = 20, b_5 = 40.$$

From (6.3), we obtain the following valid inequalities:

$$x + 15y_1 + 10y_2 \geq 15, \quad \text{i.e.,} \quad R = \{1, 2\} \tag{6.13}$$

$$x + 20y_1 + 10y_2 + 5y_3 + 5y_4 \geq 20, \quad \text{i.e.,} \quad R = \{1, 2, 4\} \tag{6.14}$$

$$x + 40y_1 + 10y_2 + 25y_3 + 5y_4 + 20y_5 \geq 40, \quad \text{i.e.,} \quad R = \{1, 2, 4, 5\}. \tag{6.15}$$

Next, we show that for a given set R, the inequality (6.3) can be strengthened.

Theorem 6.2 *Given a set $R = \{1, 2, \ldots, Q\} \subseteq \mathcal{V}$ indexed such that $b_1 \le b_2 \le \ldots \le b_Q$, define $i' = argmin\{u : u \in \mathcal{P}(i)$ and $b_u > b_{i-1}\}$ for each $i \in R$ and let $\Omega = \cup_{i=1}^{Q} \mathcal{P}(i', i)$ and $\Omega(j) = \Omega \cap \mathcal{V}(j)$. Then, the inequality (6.3) is dominated by the inequality*

$$x + \sum_{j \in \mathcal{V}_\Omega} \phi_\Omega(j) y_j \ge b_Q, \tag{6.16}$$

where $\mathcal{V}_\Omega = \cup_{i \in \Omega} \mathcal{P}(i)$ and $\phi_\Omega(j) = \min\{a_j, \sum_{i \in \Omega(j)}(b_i - b_{i-1})\}$ with $b_0 = 0$.

Proof: The validity of (6.16) follows directly from Theorem 6.1 by substituting Ω for R. Also, it is easy to see that $\phi_\Omega(j) \le \phi_R(j)$ for each $j \in \mathcal{V}_\Omega$. □

Example (continued): In the example, the inequality (6.14) is dominated by

$$x + 20y_1 + 10y_2 + 5y_3 + 3y_4 \ge 20 \tag{6.17}$$

corresponding to $\Omega = \{1, 2, 3, 4\}$ and the inequality (6.15) is dominated by

$$x + 40y_1 + 10y_2 + 25y_3 + 3y_4 + 20y_5 \ge 40 \tag{6.18}$$

corresponding to $\Omega = \{1, 2, 3, 4, 5\}$.

Next we study the polyhedral aspects of the family of inequalities (6.16). We establish facet-defining condition and we also give condition for which this family of inequalities are sufficient to describe the convex hull of X_{SDK}.

6.1.2 Facet-Defining Conditions

Theorem 6.3 *Inequality (6.16) is facet-defining for X_{SDK} if*

(1) for each $j \in \mathcal{V}_\Omega$, $a_j \ge \max\{b_i, i \in \Omega(j)\}$,

(2) for each pair $j \in \Omega$ and $r \in \mathcal{V}(j)$, $b_j + \sum_{k \in \mathcal{P}(r) \setminus \mathcal{P}(j)} a_k \ge b_r$,

(3) for each $j \in \mathcal{V} \setminus \mathcal{V}_\Omega$, there exists a $s(j) \in \mathcal{P}(j) \cap \mathcal{V}_\Omega$ such that $a_{s(j)} + \sum_{k \in \mathcal{P}(r) \setminus \mathcal{V}_\Omega} a_k \ge b_r$ for each $r \in \mathcal{P}(j) \setminus \{\mathcal{V}_\Omega \cup j\}$ and $a_{s(j)} + \sum_{k \in \mathcal{P}(r) \setminus \{\mathcal{V}_\Omega \cup j\}} a_k \ge b_r$ for each $r \in \mathcal{V}(j)$.

Proof: We construct $\dim(X) = n + 1$ linearly independent vectors belonging to X that satisfy (6.16) at equality.

We construct a vector corresponding to each of the $n+1$ variables. Let e^x and e^{y_j} be unit vectors in $\mathbb{R}^{n+1}$ corresponding to the coordinates x and y_j for $j = 1, \ldots, n$. For each $j \in \mathcal{V}_\Omega$, let $\rho(j) = \min\{k : k \in \Omega(j)\}$ and $\Phi(j) = \{i \in \Omega \cup a(j) : b_i \leq b_{\rho(j)-1}\}$. Note here, we have $b_{a(j)} \leq b_{\rho(j)-1}$ according to the definition of Ω and therefore $a(j) \in \Phi(j)$. Define $\Psi(j) = \cup_{k \in \Phi(j)} \mathcal{P}(k)$ and $\Lambda(j) = \cup_{k \in \Psi(j)} \mathcal{C}(k) \setminus \Psi(j)$. The vectors are denoted by $\{u^j\}_{j=0}^n$ and are constructed as follows.

(i) Vector u^0 corresponds to variable x and is given by

$$u^0 = b_Q e^x + \sum_{r \in \mathcal{V} \setminus \mathcal{V}_\Omega} e^{y_r}.$$

(ii) For each y_j where $j \in \mathcal{V}_\Omega$, the corresponding vector u^j is given by

$$u^j = b_{\rho(j)-1} e^x + \sum_{r \in \Lambda(j)} e^{y_r} + \sum_{r \in \mathcal{V} \setminus \mathcal{V}_\Omega} e^{y_r}.$$

(iii) For each y_j where $j \in \mathcal{V} \setminus \mathcal{V}_\Omega$, the corresponding vector u^j is given by

$$u^j = u^{s(j)} - e^j.$$

Feasibility: We need to show that $\{u^j\}_{j=0}^n$ satisfies (6.2) for each $i \in \mathcal{V}$.

(i) The vector u^0 clearly satisfies (6.2) corresponding to each $i \in \mathcal{V}_\Omega$ since the left-hand-side of (6.2) is $b_Q \geq b_i$ for each $i \in \mathcal{V}_\Omega$. Corresponding to each $i \in \mathcal{V} \setminus \mathcal{V}_\Omega$, let $\xi(i) = \text{argmax}\{b_k : k \in \mathcal{P}(i) \cap \Omega\}$ and the left-hand-side of (6.2) is

$$\begin{aligned} & b_Q + \textstyle\sum_{k \in \mathcal{P}(i) \setminus \mathcal{P}(\xi(i))} a_k \\ \geq \;& b_{\xi(i)} + \textstyle\sum_{k \in \mathcal{P}(i) \setminus \mathcal{P}(\xi(i))} a_k \\ \geq \;& b_i, \end{aligned}$$

where the first inequality follows from the fact that $b_Q \geq b_{\xi(i)}$ since $\xi(i) \in \Omega$ and the second inequality follows from condition (2).

(ii) It is easy to verify that the vector u^j corresponding to each $j \in \mathcal{V}_\Omega$ satisfies (6.2) for each $i \in \Psi(j)$. The reason is that the left-hand-side value is greater than or equal to $b_{\rho(j)-1} \geq b_i$ for each $i \in \Psi(j)$.

For each $i \in \mathcal{V}_\Omega \setminus \Psi(j)$, let $\lambda(i) = \{k : k \in \mathcal{P}(i) \cap \Lambda(j)\}$. Then, the left-hand-side of (6.2) is

$$b_{\rho(j)-1} + a_{\lambda(i)} \geq b_i$$

where the inequality follows from condition (1).

For each $i \in \mathcal{V} \setminus \mathcal{V}_\Omega$, the left-hand-side of (6.2) for the case that $\lambda(i)$ exists is

$$\begin{aligned} &\geq b_{\rho(j)-1} + a_{\lambda(i)} + \textstyle\sum_{k \in \mathcal{P}(i) \setminus \mathcal{P}(\xi(i))} a_k \\ &\geq b_{\xi(i)} + \textstyle\sum_{k \in \mathcal{P}(i) \setminus \mathcal{P}(\xi(i))} a_k \\ &\geq b_i \end{aligned}$$

where the second inequality follows from condition (1) and the last inequality follows from condition (2). Note here we can also provide a similar argument if $\lambda(i)$ does not exist.

(iii) The vector u^j corresponding to each $j \in \mathcal{V} \setminus \mathcal{V}_\Omega$ satisfies (6.2) for each $i \in \mathcal{V}_\Omega$. It follows from the fact that $s(j) \in \mathcal{V}_\Omega$ and $j \in \mathcal{V} \setminus \mathcal{V}_\Omega$. Condition (3) shows that u^j satisfies (6.2) for each $i \in \mathcal{V} \setminus \mathcal{V}_\Omega$.

Tightness: We need to show that $\{u^j\}_{j=0}^n$ satisfies (6.16) at equality.

(i) It is easy to verify that the vector u^0 satisfies (6.16) at equality.

(ii) The vector u^j for each $j \in \mathcal{V}_\Omega$ satisfies (6.16) at equality since the left-hand-side of (6.16) is

$$\begin{aligned} &= b_{\rho(j)-1} + \textstyle\sum_{r \in \Lambda(j)} \phi_\Omega(r) \\ &= b_{\rho(j)-1} + \textstyle\sum_{r \in \Lambda(j)} \sum_{i \in \Omega(r)} (b_i - b_{i-1}) \\ &= b_Q \end{aligned}$$

where the second equation follows from the fact that $a_r \geq \sum_{i \in \Omega(r)} (b_i - b_{i-1})$ for each $r \in \Lambda(j)$. This is because that $r \in \Lambda(j) \subseteq \mathcal{V}_Q$ and based on condition (i), we have

$a_r \geq \max\{b_i, i \in \Omega(r)\} \geq \sum_{i\in\Omega(r)}(b_i - b_{i-1})$. The last equation follows from the fact that $\Omega(r_1) \cap \Omega(r_2) = \emptyset$ if $r_1, r_2 \in \Lambda(j)$ and $k \in \cup_{r\in\Lambda(j)}\Omega(r)$ for each $k \in \Omega$ such that $b_k \geq b_j$.

(iii) The vector u^j for each $j \in \mathcal{V} \setminus \mathcal{V}_\Omega$ satisfies (6.16) at equality since $u^{s(j)}$ satisfies (6.16) at equality based on (ii) and $u^j = u^{s(j)} - e^j$.

Linear independence: To verify the linear independence of the $n+1$ vectors $\{u^j\}_{j=0}^n$, observe that we can first obtain $|\mathcal{V} \setminus \mathcal{V}_\Omega|$ unit vectors by getting $e^{y_j} = u^j - u^{s(j)}$ corresponding to each $j \in \mathcal{V} \setminus \mathcal{V}_\Omega$. Besides this, for each $j \in \mathcal{V}_\Omega \cup \{0\}$, let

$$v^j = u^j - \sum_{r\in\mathcal{V}\setminus\mathcal{V}_\Omega} e^{y_r}.$$

We can form a matrix where the $|\mathcal{V}_\Omega|$ vectors form the rows of the matrix and the vector corresponding to i will be placed above the vector corresponding to j if $\max\{b_k : k \in \mathcal{V}_\Omega(i)\} > \max\{b_k : k \in \mathcal{V}_\Omega(j)\}$ or $\min\{b_k : k \in \mathcal{V}_\Omega(i)\} > \min\{b_k : k \in \mathcal{V}_\Omega(j)\}$ if $\max\{b_k : k \in \mathcal{V}_\Omega(i)\} = \max\{b_k : k \in \mathcal{V}_\Omega(j)\}$. Each column corresponds to each node in $\mathcal{V}_\Omega$. Column i is placed ahead of column j if $\max\{b_k : k \in \mathcal{V}_\Omega(i)\} > \max\{b_k : k \in \mathcal{V}_\Omega(j)\}$ or $\min\{b_k : k \in \mathcal{V}_\Omega(i)\} > \min\{b_k : k \in \mathcal{V}_\Omega(j)\}$ if $\max\{b_k : k \in \mathcal{V}_\Omega(i)\} = \max\{b_k : k \in \mathcal{V}_\Omega(j)\}$. From the definition of $\Lambda(j)$ and the construction process of each u^j, we can easily observe that these vectors form a lower triangle matrix. Therefore, all these $n+1$ vectors are linearly independent. □

Example (continued): In the example, inequalities (6.13) and (6.17) are facet-defining since they satisfy all three sufficient conditions. However, it is not clear if the inequality (6.18) is facet-defining since $a_3 = 20 < \max\{b_i, i \in \Omega(3)\} = b_5 = 40$ and sufficient condition (1) is violated.

Now, we study a special class of stochastic dynamic knapsack problems that satisfy the condition $a_j \geq \max\{b_k, k \in \mathcal{V}(j)\}$ for each $j \in \mathcal{V}$. Under this condition, $\phi_\Omega(j) = \sum_{i\in\Omega(j)}(b_i - b_{i-1})$ and the inequalities (6.16) are enough to describe the convex hull of X_{SDK} and the separation algorithm runs in polynomial time.

6.1.3 Convex Hull Condition

Theorem 6.4 *If $a_j \geq \max\{b_k, k \in \mathcal{V}(j)\}$ for each $j \in \mathcal{V}$, then the family of inequalities (6.16) for all $\Omega \subseteq \mathcal{V}$, together with $0 \leq x \leq b_{\mathcal{V}}$ and $0 \leq y_j \leq 1$ for each $j \in \mathcal{V}$ describe the convex hull of X_{SDK}.*

Proof: We notice that if $a_j \geq \max\{b_k, k \in \mathcal{V}(j)\}$, then all inequalities (6.16) are valid and facet-defining based on Theorem 6.3 and $\phi_\Omega(j) = \sum_{i \in \Omega(j)}(b_i - b_{i-1})$ for each $j \in \mathcal{V}_\Omega$. We also notice that inequalities (6.2) are dominated by inequalities (6.16). We only need to show no fractional extreme points exist after adding inequalities (6.16) to X_{SDK}. In the following, we prove this by contradiction and assume $u^0 = \{x^0, y_1^0, \ldots, y_n^0\}$ is an extreme point that contains fractional elements.

First, assume there is no inequality in (6.16) such that u^0 satisfies it at equality. Without loss of generality, assume the jth element of u^0 is fractional. Then there exists two points $u^1 = u^0 + \epsilon e^{y_j}$ and $u^2 = u^0 - \epsilon e^{y_j}$ feasible for X_{SDK}. It contradicts with the assumption that u^0 is an extreme point since $u^0 = (u^1 + u^2)/2$.

If there are some inequalities in (6.16) such that u^0 satisfies them at equality, we define the set of nodes corresponding to the right-hand-side of each inequality as Φ. That is,

$$\Phi = \{j \in \mathcal{V} : x^0 + \sum_{k \in \mathcal{V}_\Omega} \phi_\Omega(k) y_k^0 = b_j \text{ for some } \Omega \subseteq \mathcal{V}\}.$$

Let $\alpha^* = \operatorname{argmin}\{b_j : j \in \Phi\}$ and Ω_j be any set corresponding to each $j \in \Phi$ such that $x^0 + \sum_{k \in \mathcal{V}_{\Omega_j}} \phi_{\Omega_j}(k) y_k^0 = b_j$. In the following, we complete the proof in several steps.

Claim 1: $\alpha^* \in \Omega_j$ for each $j \in \Phi$.

Proof: If not, there exists a node $\beta^* \in \Phi$ such that $\Omega_{\beta^*} \cap \Phi = \beta^*$. Without loss of generality, let $\Omega_{\alpha^*} = (\alpha^1, \alpha^2, \ldots, \alpha^{k_1}, \alpha^*)$ and $\Omega_{\beta^*} = (\beta^1, \beta^2, \ldots, \beta^{k_2}, \beta^*)$. According to the definition of α^*, we have $x^0 + \sum_{k \in \mathcal{V}_\Lambda} \phi_\Lambda(k) y_k^0 > b_{k_1}$ where $\Lambda = \{\alpha^1, \alpha^2, \ldots, \alpha^{k_1}\}$. We also have

$$x^0 + \sum_{k \in \mathcal{V}_\Gamma} \phi_\Gamma(k) y_k^0 > b_{k_2} \text{ where } \Gamma = \{\beta^1, \beta^2, \ldots, \beta^{k_2}\} \tag{6.19}$$

and

$$x^0 + \sum_{k \in \mathcal{V}_{\Omega_{\beta^*}}} \phi_{\Omega_{\beta^*}}(k) y_k^0 = b_{\beta^*} \tag{6.20}$$

That is,

$$\begin{aligned}
b_{\beta^*} &= x^0 + \sum_{k \in \mathcal{V}_{\Omega_{\beta^*}}} \phi_{\Omega_{\beta^*}}(k) y_k^0 \\
&= x^0 + \sum_{k \in \mathcal{V}_\Gamma} \phi_\Gamma(k) y_k^0 + (b_{\beta^*} - b_{k_2}) \sum_{k \in \mathcal{P}(\beta^*)} y_k \\
&> b_{k_2} + (b_{\beta^*} - b_{k_2}) \sum_{k \in \mathcal{P}(\beta^*)} y_k
\end{aligned}$$

where the first equation follows from (6.20) and the inequality follows from (6.19). Thus, we have

$$\sum_{k \in \mathcal{P}(\beta^*)} y_k < 1. \tag{6.21}$$

Now consider the inequality corresponding to set $\Theta = \{\alpha^1, \alpha^2, \ldots, \alpha^{k_1}, \alpha^*, \beta^*\}$ and we have

$$\begin{aligned}
& x^0 + \sum_{k \in \mathcal{V}_\Theta} \phi_\Theta(k) y_k^0 \\
= \;& x^0 + \sum_{k \in \mathcal{V}_{\Omega_{\alpha^*}}} \phi_{\Omega_{\alpha^*}}(k) y_k^0 + (b_{\beta^*} - b_{\alpha^*}) \sum_{k \in \mathcal{P}(\beta^*)} y_k^0 \\
= \;& b_{\alpha^*} + (b_{\beta^*} - b_{\alpha^*}) \sum_{k \in \mathcal{P}(\beta^*)} y_k^0 \\
< \;& b_{\beta^*},
\end{aligned}$$

where the inequality follows from (6.21). It contradicts with the fact that $x^0 + \sum_{k \in \mathcal{V}_\Theta} \phi_\Theta(k) y_k^0 \geq b_{\beta^*}$ is a valid inequality. Therefore, the conclusion holds. $\square$

Claim 2:

$$\sum_{k \in \mathcal{P}(j)} y_k^0 = 1 \tag{6.22}$$

for each $j \in \Phi \setminus \{\alpha^*\}$.

Proof: Note here, for each $j \in \mathcal{V}$ such that $b_j \geq b_{\alpha^*}$, since $\alpha^* \in \Phi$ and the inequality corresponding to the set $\Omega_j = \Omega_{\alpha^*} \cup \{j\}$ is

$$\begin{aligned} & x^0 + \sum_{k \in \mathcal{V}_{\Omega_j}} \phi_{\Omega_j}(k) y_k^0 \\ = \ & x^0 + \sum_{k \in \mathcal{V}_{\Omega_{\alpha^*}}} \phi_{\Omega_{\alpha^*}}(k) y_k^0 + (b_j - b_{\alpha^*}) \sum_{k \in \mathcal{P}(j)} y_k^0 \\ \geq \ & b_j. \end{aligned}$$

Then, we have

$$\sum_{k \in \mathcal{P}(j)} y_k^0 \geq 1 \tag{6.23}$$

for each $j \in \mathcal{V} \setminus \{\alpha^*\}$ such that $b_j \geq b_{\alpha^*}$.

We also notice that for each $j \in \Phi$ and assuming $\Omega_j = \{\alpha^1, \ldots, \alpha^r, j\}$, we have

$$\begin{aligned} b_j &= x^0 + \sum_{k \in \mathcal{V}_{\Omega_j}} \phi_{\Omega_j}(k) y_k^0 \\ &= x^0 + \sum_{k \in \mathcal{V}_{\Omega_j \setminus \{j\}}} \phi_{\Omega_j \setminus \{j\}}(k) y_k^0 + (b_j - b_{\alpha^r}) \sum_{k \in \mathcal{P}(j)} y_k^0 \\ &\geq b_{\alpha^r} + (b_j - b_{\alpha^r}) \sum_{k \in \mathcal{P}(j)} y_k^0. \end{aligned}$$

That is, we have

$$\sum_{k \in \mathcal{P}(j)} y_k^0 \leq 1 \tag{6.24}$$

for each $j \in \Phi$. Combining (6.23) and (6.24), we have

$$\sum_{k \in \mathcal{P}(j)} y_k^0 = 1 \tag{6.25}$$

for each $j \in \Phi \setminus \{\alpha^*\}$. □

Claim 3: If there is a $j \in \Phi$ such that $\alpha^* \in \mathcal{P}(j)$, then $r \in \Phi$ for each $r \in \mathcal{P}(j) \backslash \mathcal{P}(\alpha^*)$.

Proof: Since $j \in \Phi$, then according to (6.22), we have $\sum_{k \in \mathcal{P}(j)} y_k^0 = 1$. For each

$r \in \mathcal{P}(j) \setminus \mathcal{P}(\alpha^*)$, assuming $\Omega_r = \Omega_{\alpha^*} \cup \{r\}$, we have

$$\begin{aligned} b_r &= x^0 + \sum_{k \in \mathcal{V}_{\Omega_r}} \phi_{\Omega_r}(k) y_k^0 \\ &= b_{\alpha^*} + (b_r - b_{\alpha^*}) \sum_{k \in \mathcal{P}(r)} y_k^0 \\ &\leq b_{\alpha^*} + (b_r - b_{\alpha^*}) \sum_{k \in \mathcal{P}(j)} y_k^0 \\ &= b_r \end{aligned}$$

where the inequality follows from the fact that $r \in \mathcal{P}(j)$ and the equality follows the fact that $\sum_{k \in \mathcal{P}(j)} y_k^0 = 1$. The above inequality shows that $\sum_{k \in \mathcal{P}(r)} y_k^0 = \sum_{k \in \mathcal{P}(j)} y_k^0 = 1$ for each $r \in \mathcal{P}(j) \setminus \mathcal{P}(\alpha^*)$. Therefore, $r \in \Phi$ for each $r \in \mathcal{P}(j) \setminus \mathcal{P}(\alpha^*)$. Then, we have $\mathcal{P}(j) \cap \mathcal{C}(\alpha^*) \in \Phi$ and $j \in \Phi$ and follows from (6.22), we have

$$y_k^0 = 0 \text{ for each } k \in \mathcal{P}(j) \setminus (\mathcal{P}(\alpha^*) \cup \mathcal{C}(\alpha^*)). \tag{6.26}$$

Similarly, for any pair $(i, j) \in \Phi$ such that $i \in \mathcal{P}(j)$ and $i \neq \alpha^*$, we have $\sum_{k \in \mathcal{P}(i)} y_k^0 = 1$ and $\sum_{k \in \mathcal{P}(j)} y_k^0 = 1$ follows from (6.22) and then

$$y_k^0 = 0 \text{ for each } k \in \mathcal{P}(j) \setminus \mathcal{P}(i). \tag{6.27}$$

Based on the results obtained from above Step 1 and Step 2. In the following step, we show that no fractional solution points exist. □

Now we show that no fractional solution exists for three cases.

Case 1: $x^0 = 0$. Then, $x^0 + d_1 y_1^0 = d_1 y_1^0 \leq d_1$, which implies that $y_1^0 = 1$ and $\alpha^* = 1$. Based on (6.22) and $y_1^0 = 1$, we have $y_k^0 = 0$ for each $k \in \mathcal{V}_\Phi \setminus \{\alpha^*\}$ where $\mathcal{V}_\Phi = \cup_{j \in \Phi} \mathcal{P}(j)$. If there exists a $k \in \mathcal{V} \setminus \mathcal{V}_\Phi$ such that $0 < y_k^0 < 1$, there are two points $u^1 = u^0 + \epsilon e^{y_k}$ and $u^2 = u^0 - \epsilon e^{y_k}$ feasible for X_{SDK}. It contradicts with the assumption that u^0 is an extreme point since $u^0 = (u^1 + u^2)/2$.

Case 2: $x^0 \neq 0$ and $\alpha^* = 1$. If there exists a $k \in \mathcal{V} \setminus \mathcal{V}_\Phi$ such that $0 < y_k^0 < 1$, there are two points $u^1 = u^0 + \epsilon e^{y_k}$ and $u^2 = u^0 - \epsilon e^{y_k}$ feasible for X_{SDK}. It contradicts with the assumption that u^0 is an extreme point since $u^0 = (u^1 + u^2)/2$. Following (6.26), we have $y^0_k = 0$ for each $k \in \mathcal{V}_\Phi \setminus \mathcal{C}(1)$. Then, $0 < y_0 < 1$ and $y_0 + y_j = 1$ for each $j \in C(1) \cap \mathcal{V}_\Phi$. Thus, $y_j = y_k$ for any pair $(j,k) \in C(1) \cap \mathcal{V}_\Phi$. Then, any valid inequalities (6.16) with the right-hand-side value b_ℓ where $\ell \in \mathcal{C}(1) \cap \mathcal{V}_\Phi$ will be equivalent to $x^0 + b_\ell y^0_1 + (b_\ell - b_1)y^0_\ell = b_\ell$. There exists two points $u^1 = u^0 + \epsilon b_1 e^x - \epsilon e^{y_1} + \sum_{k\in\mathcal{C}(1)\cap\mathcal{V}_\Phi} \epsilon e^{y_k}$ and $u^2 = u^0 - \epsilon b_1 e^x + \epsilon e^{y_1} - \sum_{k\in\mathcal{C}(1)\cap\mathcal{V}_\Phi} \epsilon e^{y_k}$ feasible for X_{SDK}. It contradicts with the assumption that u^0 is an extreme point since $u^0 = (u^1 + u^2)/2$.

Case 3: $x^0 \neq 0$ and $\alpha^* \neq 1$. Following the same argument as in Case 1 and Case 2, there does not exist any $k \in \mathcal{V}\backslash\mathcal{V}_\Phi$ such that $0 < y^0_k < 1$. Let Φ_1 be the set of nodes in $\Phi\backslash\{\alpha^*\}$ such that no nodes in $\mathcal{V}_{\Phi_1}\backslash\{\Phi_1\cup\{\alpha^*\}\}$ belongs to set Φ. That is, $\mathcal{V}_{\Phi_1}\cap\Phi = \Phi_1\cup\{\alpha^*\}$. Similarly, let Φ_2 be the set of nodes such that $\cup_{j\in\Phi_2}\mathcal{V}(j)\cap\Phi = \Phi_2$. Based on (6.27), we have $y^0_k = 0$ for each $k \in \mathcal{V}_{\Phi_2} \setminus \mathcal{V}_{\Phi_1}$. Let $\Phi'_1 = \{j \in \Phi : \text{there exists a node } k \in \mathcal{P}(j) \text{ such that } 0 < y^0_k < 1\}$. Note here we have $\sum_{k\in\mathcal{P}(j)} y^0_k = 1$ for each $j \in \Phi'_1$ according to (6.22). Then, there exists at least one pair $(k_1, k_2) \in \mathcal{P}(j)$ for each $j \in \Phi'_1$ such that $0 < y^0_{k_1}, y^0_{k_2} < 1$.

Now, we initialize two sets $\Pi_1 = \emptyset$, $\Pi_2 = \emptyset$ and label each node $j \in \Phi'_1$ be zero (i.e., $\ell(j) = 0$ for each $j \in \Phi'_1$). Then, for each element $j \in \Phi'_1$ according to the nondecreasing sequence of b_j, we do the following steps.

(1) Let $s(j) = \text{argmin}\{k \in \mathcal{P}(j) \setminus \Pi_1 : 0 < y^0_k < 1\}$.

(2) If $\ell(j) = 0$, let $\Pi_1 = \Pi_1 \cup \{j\}$, update $\ell(j) = 1$ and $\ell(r) = 1$ for each $r \in \mathcal{V}(j) \cap \Phi'_1$. Go back to (1).

(3) Else If $\ell(j) = 1$, let $\Pi_2 = \Pi_2 \cup \{j\}$, update $\ell(j) = 2$ and $\ell(r) = 2$ for each $r \in \mathcal{V}(j) \cap \Phi'_1$. Stop.

(4) Else if $\ell(j) = 2$, stop.

We can easily observe there exists two points

$$u^1 = u^0 + \epsilon \sum_{k\in\Pi_1\cap\mathcal{P}(\alpha^*)} \phi_{\Omega_{\alpha^*}}(k) e^x - \epsilon \sum_{k\in\Pi_2\cap\mathcal{P}(\alpha^*)} \phi_{\Omega_{\alpha^*}}(k) e^x - \epsilon \sum_{k\in\Pi_1} e^{y_k} + \epsilon \sum_{k\in\Pi_2} e^{y_k}$$

and

$$u^2 = u^0 - \epsilon \sum_{k \in \Pi_1 \cap \mathcal{P}(\alpha^*)} \phi_{\Omega_{\alpha^*}}(k) e^x + \epsilon \sum_{k \in \Pi_2 \cap \mathcal{P}(\alpha^*)} \phi_{\Omega_{\alpha^*}}(k) e^x + \epsilon \sum_{k \in \Pi_1} e^{y_k} - \epsilon \sum_{k \in \Pi_2} e^{y_k}$$

feasible for X_{SDK}. It contradicts with the assumption that u^0 is an extreme point since $u^0 = (u^1 + u^2)/2$. □

Remark 6.3 *When the problem has only one scenario, i.e., $|\Omega| = 1$, the convex hull result can be derived based on the convex hull result developed in Barany et al. [11].*

Remark 6.4 *When the problem has only two periods, i.e., $t(j) \leq 2$ for each $j \in \mathcal{V}$, the convex hull result can be derived from Günlük & Pochet [48].*

Example (continued): In the example, if we modify the coefficients to $a_1 = a_2 = a_3 = a_4 = a_5 = 40$, then inequalities (6.16) corresponding to $\Omega = \{1\}, \{1,2\}, \{1,3\}, \{1,3,4\}, \{1,3,5\}, \{1,2,3\}, \{1,2,3,4\}, \{1,2,3,5\}, \{1,3,4,5\}, \{1,2,3,4,5\}$ together with $0 \leq x \leq 40$ and $0 \leq y_1, \ldots, y_5 \leq 1$ describe the convex hull of all feasible solutions.

Theorem 6.5 *For the general stochastic scenario tree case, an inequality generated by an arbitrary sequence of pairing operations is dominated by some inequality obtained by a linear combination of all the inequalities (6.16) generated by our sequential pairing approach.*

Proof: When $a_j \geq \max\{b_k, k \in \mathcal{V}(j)\}$, according to the proof for the convex hull case, we know that the inequalities generated by any arbitrary sequence will be dominated by some inequality obtained by the linear combination of some inequalities generated by our sequential pairing approach. In other words, if A^0 is some inequality generated by some arbitrary sequence. Then, there exists a set of inequalities $A^1, A^2, \ldots, A^K$ generated by sequential pairing such that we have

$$A^0 \preceq \sum_{i=1}^{K} \lambda_i A^i, \text{ where } \lambda_i \geq 0 \text{ and } \sum_{i=1}^{K} \lambda_i = 1. \tag{6.28}$$

Let $\phi^i_\Omega(j)$ be the coefficient of y_j in A^i for the case $a_j \geq \max\{b_k, k \in \mathcal{V}(j)\}$. Then, we have

$$\phi^0_\Omega(j) \geq \sum_{i=1}^{K} \lambda_i \phi^i_\Omega(j) \tag{6.29}$$

corresponding to each node $j \in \mathcal{V}$.

For the general case, let $\hat{\phi}^i_\Omega(j)$ be the coefficient of y_j in A^i and we have $\hat{\phi}^i_\Omega(j) = \min\{a_j, \phi^i_\Omega(j)\}$. Then, we only need to show that

$$\min\{a_j, \phi^0_\Omega(j)\} \geq \sum_{i=1}^{K} \lambda_i \min\{a_j, \phi^i_\Omega(j)\}, \text{ where } \lambda_i \geq 0 \text{ and } \sum_{i=1}^{K} \lambda_i = 1. \tag{6.30}$$

If $a_j \geq \phi^0_\Omega(j)$, then we have

$$\min\{a_j, \phi^0_\Omega(j)\} = \phi^0_\Omega(j) \geq \sum_{i=1}^{K} \lambda_i \phi^i_\Omega(j) \geq \sum_{i=1}^{K} \lambda_i \min\{a_j, \phi^i_\Omega(j)\}$$

where the first inequality follows from (6.29).

Else if $a_j \leq \phi^0_\Omega(j)$, then

$$\min\{a_j, \phi^0_\Omega(j)\} = a_j \geq \sum_{i=1}^{K} \lambda_i a_j \geq \sum_{i=1}^{K} \lambda_i \min\{a_j, \phi^i_\Omega(j)\}.$$

Therefore, we have that (6.30) holds. □

6.1.4 Separation Algorithms

Let C be the set of inequalities (6.16) generated by sequential pairing. Note that $|C|$ is exponential in the size of the input. Although we do not know a general polynomial-time separation algorithm for the complete family of inequalities (6.16), here we present a polynomial time separation algorithm for finding a most violated inequality in C if one exists for the case that $a_j \geq \max\{b_k, k \in \mathcal{V}(j)\}$ for each $j \in \mathcal{V}$. The algorithm is based on solving shortest path problems on a directed graph G with nodes $\mathcal{N} = \{0, 1, \ldots, N\}$ and arcs (i, j) for all i and $b_j > b_i$. Given a point x^*, the separation problem of determining whether there exists a violated pairing inequality can be reduced to finding a shortest path from node 0 to node k for each $k \in \mathcal{V}$ where the length of arc (i, j) is given by $\sum_{r \in \mathcal{P}(j)} (b_j - b_i) x^*_r$. This is true because a path $P = (0, i_1, i_2, \ldots, i_k)$ in G corresponds to a valid inequality in the form (6.3) with $R = \{0, i_1, i_2, \ldots, i_k\}$ since the length of the path is equal to the left-hand side of the inequality. Therefore, there is a violated inequality with right-hand side b_{i_k} if and only if the length of a shortest path from 0 to k is less than b_{i_k}. Using Dijkstra's algorithm the separation problem can be solved in $O(N^2)$ time and we can find as many as N violated inequalities from the shortest paths from 0 to k for each $k \in \mathcal{V}$. When the condition that $a_j \geq \max\{b_k, k \in \mathcal{V}(j)\}$ does not hold, we use the above algorithm as a heuristic. After obtaining a shortest path according to the above algorithm, we insert the inequality (6.16) corresponding to this path.

Theorem 6.6 *If $a_j \geq b_{r_k}$ $\forall r_k \in \Omega(j)$, then there exists a polynomial-time separation algorithm.*

6.2 Applications

In this section, we describe some applications of the inequalities developed in this chapter to stochastic lot-sizing problems.

6.2.1 Stochastic Discrete Lot-Sizing Problem

The stochastic discrete lot-sizing problem is the stochastic generalization of the discrete lot-sizing problem [42]. In the discrete lot-sizing problem, we have demands d_i and capacities a_i in periods $i = 1, \dots, n$.. Production in period i is either 0 or a_i and the objective is to minimize total production and inventory holding cost. The feasible region of the discrete lot-sizing problem can be described as

$$X_{\text{SDLS1}} = \{(s, y) \in \mathbb{R}_+^{n+1} \times \mathcal{B}^n : s_{a(j)} + a_j y_j \geq d_j + s_j \text{ for each } j \in \mathcal{V}\}$$

where s_j represents the inventory at the end of period $t(j)$ and y_j is the indicator variable for a production set-up in period $t(j)$ corresponding to the state defined by node j.

After eliminating s_j from X_{SDLS}, we obtain the equivalent feasible region description when the unit production cost and inventory cost are nonnegative given by

$$X_{\text{SDLS}} = \{(x, y) \in \mathbb{R}_+^1 \times \mathcal{B}^n : x_0 + \sum_{k \in \mathcal{P}(j)} a_k y_k \geq \sum_{k \in \mathcal{P}(j)} d_k \text{ for each } j \in \mathcal{V}\}. \tag{6.31}$$

Let $b_j = \sum_{k \in \mathcal{P}(j)} d_k$, then the above formulation is equivalent to (6.2) for X_{SDK}. The analyses of the valid inequalities and convex hull results are the same as for X_{SDK}.

6.2.2 Stochastic Uncapacitated Lot-Sizing Problem

As described in Chapter 4, the feasible region of the stochastic uncapacitated lot-sizing problem can be given by

$$X_{\text{SLS}} = \{(s,x,y) \in \mathbb{R}_+ \times \mathbb{R}^n_+ \times \mathcal{B}^n : s_0 + \sum_{k\in\mathcal{P}(j)} x_k \geq d_{1j},\ x_j \leq M_j y_j \text{ for each } j \in \mathcal{V}\}. \quad (6.32)$$

Theorem 6.7 *If* $s_0 + \sum_{j\in\mathcal{V}_\Omega} \pi_j y_j \geq \pi_0$ *is a valid inequality for* X_{SLS} *and* $S_\Omega \subseteq \mathcal{V}_\Omega$*, then*

$$s_0 + \sum_{j\in S_\Omega} x_j + \sum_{j\in\overline{S}_\Omega} \pi_j y_j \geq \pi_0 \quad (6.33)$$

where $\overline{S}_\Omega = \mathcal{V}_\Omega \setminus S_\Omega$*, is a valid inequality for* X_{SLS}*.*

Proof: Consider a point $(s^*, x^*, y^*) \in X_{\text{SLS}}$. Now we construct a point $(\hat{x}, \hat{y}, \hat{s})$ defined by $\hat{x}_j = x^*_j$ and $\hat{y}_j = y^*_j$ for each $j \in \mathcal{V} \setminus S_\Omega$, $\hat{x}_j = \hat{y}_j = 0$ for each $j \in S_\Omega$, $\hat{s}_0 = s^*_0 + \sum_{j\in S_\Omega} x^*_j$, and $\hat{s}_j = \hat{s}_{a(j)} + \hat{x}_j - d_j$ for each $j \in \mathcal{V}$. Then, $(\hat{s}, \hat{x}, \hat{y}) \in X_{\text{SLS}}$. This is because all production in the nodes S_Ω has been inserted into the initial stock and it is easy to verify that $(\hat{s}, \hat{x}, \hat{y})$ satisfies (6.32) as long as (s^*, x^*, y^*) satisfies (6.32). Then, $\hat{s}_0 + \sum_{j\in\mathcal{V}_\Omega} \pi_j \hat{y}_j \geq \pi_0$, which implies that $s^*_0 + \sum_{j\in S_\Omega} x^*_j + \sum_{j\in\overline{S}_\Omega} \pi_j y^*_j \geq \pi_0$. Therefore, the inequality is valid for X_{SLS}. $\Box$

Theorem 6.8 *Given any set* $\Omega \subseteq \mathcal{V}$ *and any subset* $S_\Omega \subseteq \mathcal{V}_\Omega$*, let* $\phi_\Omega(j) = \sum_{i\in\Omega(j)} (b_i - b_{i-1})$ *with* $b_0 = 0$ *for each* $j \in \mathcal{V}$*. Then the inequality*

$$s_0 + \sum_{j\in S_\Omega} x_j + \sum_{j\in\overline{S}_\Omega} \phi_\Omega(j) y_j \geq M_\Omega(1), \quad (6.34)$$

called a (Ω, S_Ω) *inequality where* $\overline{S}_\Omega = \mathcal{V}_\Omega \setminus S_\Omega$, $M_\Omega(1) = \max\{d_{1k} : k \in \mathcal{V}_\Omega\}$ *is valid for* X_{SLS}*.*

Proof: Consider the formulation in (6.32) and relax x_j to $M_j y_j$ in the formulation. We have the following inequalities

$$s_0 + \sum_{k\in\mathcal{P}(j)} M_k y_k \geq d_{1j} \text{ for each } j \in \mathcal{V}$$

valid for X_{SLS}. After applying the sequential pairing scheme as for the stochastic dynamic knapsack problem, we obtain the inequality

$$s_0 + \sum_{j \in \mathcal{V}_\Omega} \phi_\Omega(j) y_j \geq M_\Omega(1) \tag{6.35}$$

where $\phi_\Omega(j) = \sum_{i \in \Omega(j)} (b_i - b_{i-1})$ with $b_0 = 0$ since M_j is large, which is valid for X_{SLS}. By Theorem 6.7, the inequality

$$s_0 + \sum_{j \in S_\Omega} x_j + \sum_{j \in \overline{S}_\Omega} \phi_\Omega(j) y_j \geq M_\Omega(1) \tag{6.36}$$

is valid for X_{SLS}. □

Remark 6.5 *The inequalities (6.35) is a special case of the inequality (6.36) with* $S_\Omega = \emptyset$*.*

Proposition 6.1 *The* $(\mathcal{Q}, S_\mathcal{Q})$ *inequalities developed in Chapter 4 are special cases of the* (Ω, S_Ω) *inequalities described in (6.34).*

Proof: The proof follows from the fact that $\mathcal{Q} \subseteq \Omega$ and $\Delta_\mathcal{Q}(j)$ defined in Chapter 4 is the same as $\phi_\mathcal{Q}(j)$ defined in (6.34) for a common set $\mathcal{Q} \subseteq \mathcal{V}$. □

From these results it follows that:

Theorem 6.9 *For the two-period stochastic uncapacitated lot-sizing problem, let* $\Omega = \{0, 1', 2', \ldots, r'\} \subseteq \mathcal{V}$ *be indexed such that* $d_{1'} \leq d_{2'} \leq \ldots \leq d_{r'}$*. The inequalities*

$$x_0 + x_{1'} + \sum_{j' \in \Omega \setminus \{0,1'\}} (d_{j'} - d_{j-1'}) y_{j'} \geq d_{0r'} \tag{6.37}$$

and

$$x_0 + d_{1'} y_{1'} + \sum_{j' \in \Omega \setminus \{0,1'\}} (d_{j'} - d_{j-1'}) y_{j'} \geq d_{0r'} \tag{6.38}$$

together with $x_j \geq 0$, $y_j \leq 1$ *and* $x_j \leq M_j y_j$ *for each* $j \in \mathcal{V}$ *are enough to describe the convex hull of* X_{SLS}*.*

6.2.3 Stochastic Capacitated Lot-Sizing Problem

The stochastic capacitated lot-sizing problem is the stochastic extension of the deterministic capacitated lot-sizing problem. The deterministic capacitated lot-sizing problem is well studied, for example see [65, 81, 82, 8]. Compared to the uncapacitated version, the production quantity for the capacitated lot-sizing problem has an upper bound a_j in each time period j. The feasible region for the stochastic capacitated lot-sizing problem is

$$X_{\text{SCLS}} = \{(s, x, y) \in \mathbb{R}_+ \times \mathbb{R}_+^n \times \mathcal{B}^n : s_0 + \sum_{k \in \mathcal{P}(j)} x_k \geq d_{1j},\ x_j \leq a_j y_j \text{ for each } j \in \mathcal{V}\}. \quad (6.39)$$

Theorem 6.10 *Given any set $\Omega \subseteq \mathcal{V}$ and any subset $S_\Omega \subseteq \mathcal{V}_\Omega$, let $\phi_\Omega(j) = \min\{a_j, \sum_{i \in \Omega(j)} (b_i - b_{i-1})\}$ with $b_0 = 0$ for each $j \in \mathcal{V}$. Then, the inequality*

$$s_0 + \sum_{j \in S_\Omega} x_j + \sum_{j \in \overline{S}_\Omega} \phi_\Omega(j) y_j \geq M_\Omega(1), \quad (6.40)$$

where $\overline{S}_\Omega = \mathcal{V}_\Omega \setminus S_\Omega$ and $M_\Omega(1) = \max\{d_{1k} : k \in \mathcal{V}_\Omega\}$, is valid for X_{SCLS}.

Proof: Similar to the proof of Theorem 6.8, we can relax x_j to $a_j y_j$ in the formulation and apply sequential pairing to obtain the following inequality

$$s_0 + \sum_{j \in \mathcal{V}_\Omega} \phi_\Omega(j) y_j \geq M_\Omega(1) \quad (6.41)$$

where $\phi_\Omega(j) = \min\{a_j, \sum_{i \in \Omega(j)} (b_i - b_{i-1})\}$ with $b_0 = 0$, which is valid for X_{SCLS}. We also notice that Theorem 6.7 is also valid for the stochastic capacitated lot-sizing problem. Therefore, inequality (6.40) is valid for X_{SCLS}. □

Similar to Theorem 6.9, we obtain

Theorem 6.11 *For the two-period stochastic capacitated lot-sizing problem, let $\Omega = \{0, 1', 2', \ldots, r'\} \subseteq \mathcal{V}$ be indexed such that $d_{1'} \leq d_{2'} \leq \ldots \leq d_{r'}$. Define $\overline{d}_i = d_{0i} - \max\{d_0, \Delta\}$ where $\Delta = \max\{d_{0j} - a_j : j \in \Omega\}$ and $\rho' = \text{argmin}\{d_{0j} : j \in \Omega \text{ and } d_{0j} > \Delta\}$. Then the following inequalities*

$$x_0 + x_{\rho'} + \sum_{j' \in \Omega \setminus \{0, \ldots, (\rho-1)'\}} (\overline{d}_{j'} - \overline{d}_{(j-1)'}) y_{j'} \geq d_{0r'} \quad (6.42)$$

and

$$x_0 + \overline{d}_{\rho'} y_{\rho'} + \sum_{j' \in \Omega \setminus \{0,\ldots,(\rho-1)'\}} (\overline{d}_{j'} - \overline{d}_{(j-1)'}) y_{j'} \geq d_{0r'} \tag{6.43}$$

together with $x_j \geq 0$, $y_j \leq 1$ and $x_j \leq a_j y_j$ for each $j \in \mathcal{V}$ are sufficient to describe the convex hull of X_{SCLS}.

Proof: First we observe that $x_0 \geq \max\{d_0, \Delta\}$, where $\Delta = \max\{d_{0j} - a_j : j \in \Omega\}$. Note that $x_0 \geq d_0$. In order to keep feasibility for each node $j \in \mathcal{C}(0)$, we have $x_0 \geq \Delta$. Otherwise, if $x_0 < \Delta$, let $j^* = \text{argmax}\{d_{0j} - a_j : j \in \Omega\}$ and then $x_0 + a_{j^*} y_{j^*} < \Delta + a_{j^*} = d_{0j^*}$, which is contradiction.

If $d_0 \geq \Delta$, then $\rho = 1$ and $d_j = d_{0j} - d_0 \leq d_{0j} - \Delta \leq a_j$ for each $j \in \mathcal{C}(0)$ since $\Delta = \max\{d_{0j} - a_j : j \in \Omega\}$. We can tighten a_j to d_j in the formulation. Then the problem is equivalent to the two-period stochastic uncapacitated lot-sizing problem. According to Theorem 6.9, the inequalities (6.42) and (6.43) are valid based on inequalities (6.37) and (6.38) and define the convex hull of X_{SCLS}.

If $d_0 < \Delta$, then the problem is equivalent to the problem such that $\overline{d}_0 = \Delta$ and $\overline{d}_j = d_{0j} - \Delta$ for each $j \in \mathcal{C}(0)$. According to the definition of Δ, we have $\overline{d}_j \leq a_j$ for each $j \in \mathcal{C}(0)$. Then, we can tighten a_j to $\overline{d}_j$ for each node $j \in \mathcal{C}(0)$. According to Theorem 6.9, the inequalities (6.42) and (6.43) are valid based on inequalities (6.37) and (6.38) and define the convex hull of X_{SCLS}. □

6.3 Computational Experiments

In this section we present some numerical results to demonstrate the computational effectiveness of the inequalities generated by our pairing scheme on randomly generated instances of mixed-integer programs with stochastic dynamic knapsack constraints. Specifically, we perform the experiments for multi-item stochastic capacitated lot-sizing problems. All computations have been carried out on a Linux workstation with dual 2.4 GHz Intel Xeon processors and 2 GB RAM using CPLEX 8.1.

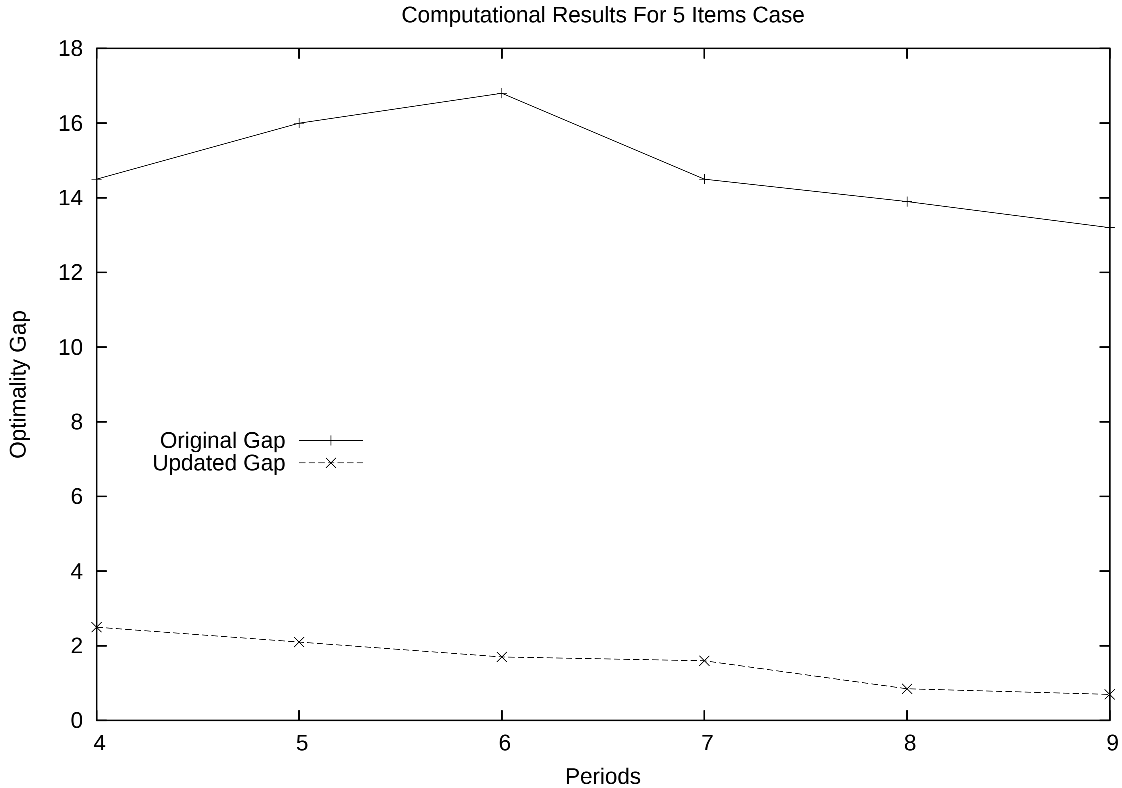

Figure 6: Computational results for 5 items case

Given a set of items (i.e., denoted as set $\mathcal{I}$), let c_j be the capacity corresponding to node j in the tree. The multi-item stochastic capacitated lot-sizing problem can be formulated as

$$
\begin{aligned}
(\text{MCSLS}) : \min \quad & \sum_{i \in \mathcal{V}} p_i \left(\sum_{k \in \mathcal{I}} \alpha_i x_{ik} + \beta_i y_i \right) \\
& \sum_{j \in \mathcal{P}(i)} x_{jk} \geq d_{0i}^k \qquad \forall i \in \mathcal{V}, k \in \mathcal{I} \\
& \sum_{k \in I} x_{ik} \leq c_i y_i \qquad \forall i \in \mathcal{V} \\
& y_i \in \{0, 1\}, x_i \geq 0 \qquad \forall i \in \mathcal{V}.
\end{aligned}
$$

In the computational experiments, we randomly generate two sets of instances that contain 5 items and 10 items respectively. For each set, we generate instances that have 4 to 9 periods and have two branches for each node with 0.5 probability each. Demands are uniformly distributed in the interval $[10, 100]$ and setup costs are uniformly distributed in the interval $[10, 200]$. The inventory costs are uniformly distributed in the interval $[0.01, 0.05]$.

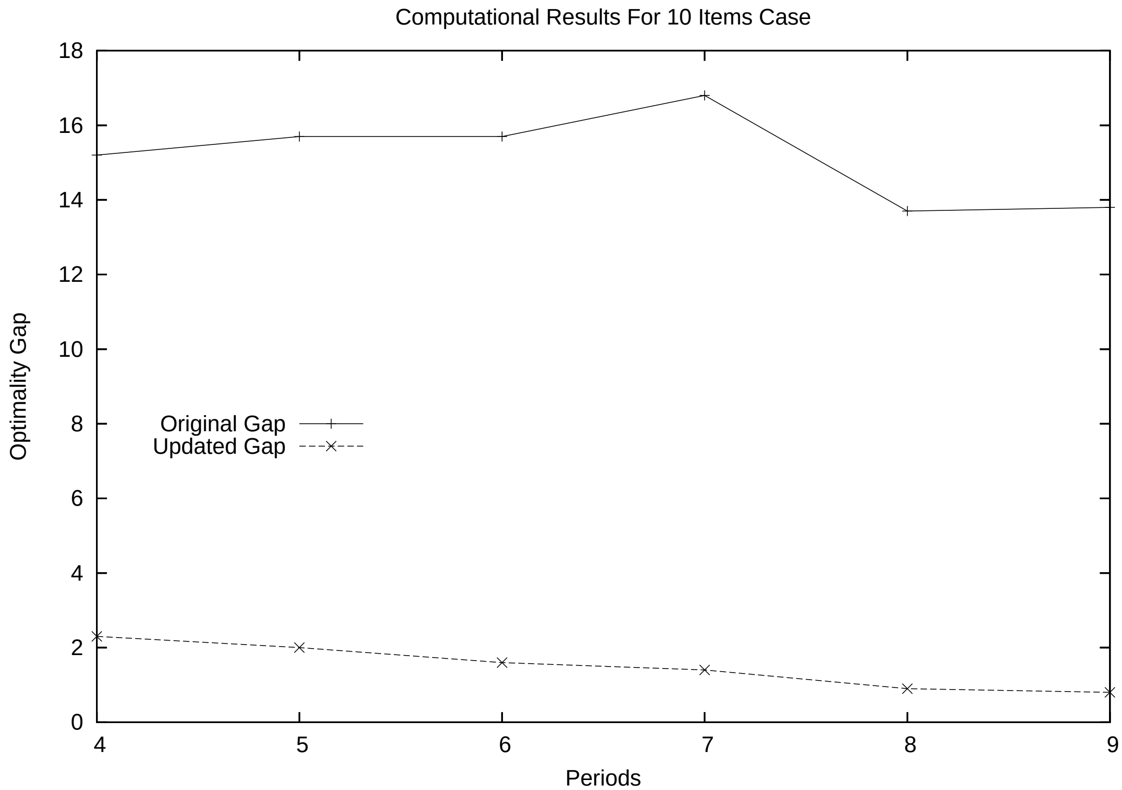

Figure 7: Computational results for 10 items case

For each setting, we test 5 replications and evaluate the average value. The gap percentage is calculated as Gap $= \frac{\text{UB}-\text{LB}}{\text{UB}} \times 100$, where "UB" is the optimal objective value in our instances and "LB" is the linear program lower bound. The gaps are shown in Figure 6 and 7 for 5 items and 10 items respectively. In these figures, the original gap is the linear program relaxation without adding any inequalities as cuts while the updated gap represents the optimality gap after adding inequalities generated by our approach as cuts at the root node. We observe that the optimality gap reduces significantly from more than 14% to less than 2.5% after adding our inequalities as cuts.

CHAPTER 7

CONCLUSION AND FUTURE RESEARCH

In this thesis, we used stochastic lot-sizing problems to perform polyhedral studies of stochastic integer programs. We developed methods to solve the stochastic uncapacitated lot-sizing problem through a polyhedral study of its mixed integer programming formulation. In particular, we developed strong valid inequalities for the problem, and incorporated them within a branch-and-cut scheme. Then, we analyzed a systematic approach to generate these inequalities for more general stochastic integer programming problems. We introduced a *pairing* scheme to generate a new valid inequality based on two existing valid inequalities and especially, we analyzed paring sequences leading to non-dominated inequalities. This idea was then used to develop new valid inequalities for various classes of stochastic and deterministic integer programs. In this section, we briefly discuss possible future research directions.

Stochastic lot-sizing problems: In this thesis, we developed the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequalities for the stochastic uncapacitated lot-sizing problem. A number of issues deserve further investigation.

(i) The complexity of separating the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequalities for the two-period case is known to be polynomial. However, the complexity is unknown for more than two periods.

(ii) The $(\mathcal{Q}, S_{\mathcal{Q}})$ inequalities are sufficient to describe the convex hull of solutions of the two-period stochastic uncapacitated lot-sizing problem. However, in general, we know that the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequalities are not sufficient to describe the convex hull of solutions for multi-period problems. We need to investigate relationship between problem parameters such that the $(\mathcal{Q}, S_{\mathcal{Q}})$ inequalities are sufficient to describe the convex hull or provide integral solutions. For general problem parameter settings, other inequalities need to be developed with particular consideration given to lifting the continuous variables.

(iii) In this thesis, we applied the pairing scheme to generate valid inequalities for the stochastic capacitated lot-sizing problem. Some preliminary computational work was done on evaluating the performance of adding these inequalities as cuts. We should study variations individually including start-up cost, set-up time and backlogging.

Pairing inequalities: We introduced a pairing scheme to generate new valid inequalities based on existing valid inequalities. We identified useful pairing sequences for a number of constraint structures and introduced some important applications of the pairing scheme. Even though this procedure can provide the convex hull description of feasible solutions for some special cases, further research needs to be explored for more general cases.

(i) For the new family of inequalities obtained by the pairing scheme, we only provided a sufficient condition such that this family of inequalities is facet-defining. This condition is strong. As a future subject, it will be interesting to discover necessary conditions for these inequalities to be facet-defining. We also need to study the complexity of separation in general.

(ii) Although the new family of inequalities is sufficient to describe the convex hull of all feasible solutions for some special cases, more work needs to be done for general cases. It will be interesting to see if sequence independent lifting or lifting continuous variables yields more strong valid inequalities.

There are other interesting research topics that can be directly extended from this thesis. We may need to evaluate the performance of the branch-and-cut algorithm after adding these inequalities in some complicated application models, such as capacity expansion and facility location problems. It is also important to develop efficient heuristic algorithms to provide good upper bounds so as to speed up the branch-and-cut algorithm.

In general, our research is a first step in the polyhedral study of deterministic equivalent formulations of stochastic integer programs. These ideas can be applied to various classes of stochastic integer programming problems and can also be combined with decomposition algorithms for solving large-scale stochastic integer programming problems.

REFERENCES

[1] Aggarwal, A. and Park, J. K., "Improved algorithms for economic lot size problems," *Operations Research*, vol. 41, pp. 549–571, 1993.

[2] Aghezzaf, E. H. and Wolsey, L. A., "Modelling piecewise linear concave costs in a tree partitioning problem," *Discrete Applied Mathematics*, vol. 50, pp. 101–109, 1994.

[3] Ahmed, S., King, A. J., and Parija, G., "A multi-stage stochastic integer programming approach for capacity expansion under uncertainty," *Journal of Global Optimization*, vol. 26, pp. 3–24, 2003.

[4] Ahmed, S. and Sahinidis, N. V., "An approximation scheme for stochastic integer programs arising in capacity expansion," *Operations Research*, vol. 51, pp. 461–471, 2003.

[5] Ahmed, S., Tawarmalani, M., and Sahinidis, N. V., "A finite branch and bound algorithm for two-stage stochastic integer programs," *Mathematical Programming*, vol. 100, pp. 355–377, 2004.

[6] Alonso-Ayuso, A., Escudero, L. F., Garin, A., Ortuno, M. T., and Perez, G., "An approach for strategic supply chain planning under uncertainty based on stochastic 0-1 programming," *Journal of Global Optimization*, vol. 26, pp. 97–124, 2003.

[7] Ariyawansa, K. A. and Hudson, D. D., "Performance of a benchmark parallel implementation of the van Slyke and Wets algorithm for two-stage stochastic programs on the sequent/balance," *Concurrency: Practice and Experience*, vol. 3, pp. 109–128, 1990.

[8] Atamtürk, A. and Münoz, J. C., "A study of the lot-sizing polytope," *Mathematical Programming*, 2004. to appear.

[9] Atamtürk, A., Nemhauser, G. L., and Savelsbergh, M. W. P., "The mixed vertex packing problem," *Mathematical Programming*, vol. 89, pp. 35–53, 2000.

[10] Balas, E., "Disjunctive programming: Cutting planes from logical conditions," in *Nonlinear Programming 2* (Mangasarian, O. L., ed.), pp. 279–312, Academic Press, 1975.

[11] Barany, I., van Roy, T., and Wolsey, L. A., "Strong formulations for multi-item capacitated lot sizing," *Management Science*, vol. 30, pp. 1255–1262, 1984.

[12] Barany, I., van Roy, T., and Wolsey, L. A., "Uncapacitated lot-sizing: The convex hull of solutions," *Mathematical Programming Study*, vol. 22, pp. 32–43, 1984.

[13] Beale, E. M. L., "On minimizing a convex function subject to linear inequalities," *Journal of the Royal Statistical Society, Series B*, vol. 17, pp. 173–184, 1955.

[14] Beale, E. M. L., Dantzig, G. B., and Watson, R. D., "A first-order approach to a class of multi-time-period stochastic programming problems," *Mathematical Programming Study*, vol. 27, pp. 103–117, 1986.

[15] Benders, J. F., "Partitioning procedures for solving mixed-variable programming problems," *Numerische Mathemstik*, vol. 4, pp. 238–252, 1962.

[16] Beraldi, P. and Ruszczyński, A., "A branch and bound method for stochastic integer problems under probabilistic constraints," *Optimization Methods and Software*, vol. 17, pp. 359–382, 2002.

[17] Birge, J. R., "Decomposition and partitioning methods for multi-stage stochastic linear programming," *Operations Research*, vol. 33, pp. 989–1007, 1985.

[18] Birge, J. R. and Louveaux, F. V., "A multicut algorithm for two-stage linear programs," Technical Report, Department of IOE, University of Michigan, Ann Arbor, MI, 1985.

[19] Birge, J. R. and Louveaux, F. V., *Introduction to Stochastic Programming.* Springer-Verlag, New York, 1997.

[20] Carøe, C. C., *Decomposition in stochastic integer programming.* PhD thesis, University of Copenhagen, 1998.

[21] Carøe, C. C. and Schultz, R., "Dual decomposition in stochastic integer programming," *Operations Research Letters*, vol. 24, pp. 37–45, 1999.

[22] Carøe, C. C. and Tind, J., "A cutting plane approach to mixed 0-1 stochastic integer programs," *European Journal of Operational Research*, vol. 101, pp. 306–316, 1997.

[23] Carøe, C. C. and Tind, J., "L-shaped decomposition of two-stage stochastic programs with integer recourse," *Mathematical Programming*, vol. 83, pp. 139–152, 1998.

[24] Chvátal, V., "Edmonds polytopes and a hierarchy of combinatorial problems," *Discrete Mathematics*, vol. 4, pp. 305–327, 1973.

[25] Chvátal, V., "Edmonds polytopes and weekly hamiltonian graphs," *Mathematical Programming*, vol. 5, pp. 29–40, 1973.

[26] Chvátal, V., *Linear Programming.* W H Freeman & Co., 1983.

[27] Cook, S. A., "The complexity of theorem-proving procedures," in *Proceedings of the Third annual ACM symposium on Theory of Computing* (Harrison, M. A., Banerji, R. B., and Ullman, J. D., eds.), pp. 151–158, Association for Computing Machinery, New York, 1971.

[28] Cook, W., Kannan, R., and Schrijver, A. J., "Chvátal closures for mixed integer programming problems," *Mathematical Programming*, vol. 47, pp. 155–174, 1990.

[29] Crowder, H., Johnson, E., and Padberg, M., "Solving large-scale zero-one linear programming problems," *Operations Research*, vol. 31, pp. 803–834, 1983.

[30] Dantzig, G. B., "Maximization of a linear function of variables subject to linear inequalities," in *Activity Analysis of Production and Allocation* (Koopmans, T. C., ed.), pp. 339–347, John Wiley and Sons, 1951.

[31] Dantzig, G. B., "Linear programming under uncertainty," *Management Science*, vol. 1, pp. 197–206, 1955.

[32] Dantzig, G. B., "Planning under uncertainty using parallel computing," *Annals of Operations Research*, vol. 14, pp. 1–16, 1988.

[33] Dantzig, G. B., "Linear programming," *Operations Research*, vol. 50, pp. 42–47, 2002.

[34] Dantzig, G. B. and Madansky, M., "On the solution of two-staged linear programs under uncertainty," in *Proceedings of the 4th Berkeley Symposium on Mathematical Statistics and Probability I* (Neymann, J., ed.), pp. 165–176, Springer Verlag, Berlin, 1961.

[35] Dantzig, G. B. and Wolfe, P., "The decomposition principle for linear programs," *Operations Research*, vol. 8, pp. 110–111, 1960.

[36] Dash Optimization, *XPRESS-MP Users Guide, Release 13*, 2001.

[37] Dentcheva, D. and Romisch, W., "Optimal power generation under uncertainty via stochastic programming," in *Stochastic Programming Methods and Technical Applications. Lecture Notes in Economics and Mathematical Systems* (Marti, K. and Kall, P., eds.), vol. 458, pp. 22–56, Springer Verlag, Berlin, 1998.

[38] Ermoliev, Y., "Stochastic quasi-gradient methods and their applications to systems optimization," *Stochastics*, vol. 9, pp. 1–36, 1983.

[39] Ermoliev, Y., "Stochastic quasi-gradient methods," in *Numerical Techniques for Stochastic Optimization* (Ermoliev, Y. and Wets, R. J.-B., eds.), pp. 141–186, Springer Verlag, Berlin, 1988.

[40] Ermoliev, Y., "Stochastic quasigradient methods and their implementation," in *Numerical Techniques for Stochastic Optimization* (Ermoliev, Y. and Wets, R. J.-B., eds.), pp. 313–351, Springer Verlag, Berlin, 1988.

[41] Federgruen, A. and Tzur, M., "A simple forward algorithm to solve general dynamic lot sizing models with n periods in $\o(n \log n)$ or $\o(n)$ time," *Management Science*, vol. 37, pp. 909–925, 1991.

[42] Fleischmann, B., "The discrete lot-sizing and scheduling problem," *European Journal of Operational Research*, vol. 44, pp. 337–348, 1990.

[43] Gassmann, H., "Mslip: a computer code for the multi-stage stochastic linear programming problem," *Mathematical Programming*, vol. 47, pp. 407–423, 1990.

[44] Gomory, R. E., "Outline of an algorithm for integer solutions to linear programs," *Bulletin of the American Mathematical Society*, vol. 64, pp. 275–278, 1958.

[45] Guan, Y., Ahmed, S., Miller, A. J., and Nemhauser, G. L., "On formulations of the stochastic uncapacitated lot-sizing problem," *Operations Research Letters*. To appear.

[46] Guan, Y., Ahmed, S., and Nemhauser, G. L., "Sequential pairing of mixed integer inequalities," TLI-04-04, School of Industrial and Systems Engineering, Georgia Institute of Technology, 2004.

[47] Guan, Y., Ahmed, S., Nemhauser, G. L., and Miller, A. J., "A branch-and-cut algorithm for the stochastic uncapacitated lot-sizing problem," *Mathematical Programming*. To appear.

[48] Günlük, O. and Pochet, Y., "Mixing MIR inequalities for mixed integer programs," *Mathematical Programming*, vol. 90, pp. 429–457, 2001.

[49] Haugen, K. K., Løkketangen, A., and Woodruff, D. L., "Progressive hedging as a meta-heuristic applied to stochastic lot-sizing," *European Journal of Operational Research*, vol. 132, pp. 103–109, 2001.

[50] Higle, J. L., Lowe, W., and Odio, R., "Conditional stochastic decomposition: an algorithmic interface for optimization/simulation," Working Paper 90–009, Department of Systems and Industrial Engineering, The University of Arizona, Tucson, AZ, 1990.

[51] Higle, J. L. and Sen, S., "Statistical verification of optimality conditions for stochastic programs with recourse," *Annals of Operations Research*, vol. 30, pp. 215–240, 1989.

[52] Higle, J. L. and Sen, S., "Stochastic decomposition: an algorithm for two-stage linear programs with recourse," *Mathematics of Operations Research*, vol. 16, pp. 650–669, 1989.

[53] Hiller, R. S. and Eckstein, J., "Designing fixed income portfolios using massively parallel benders decomposition," Working Paper 91–025, Harvard Business School, Cambridge, MA, 1990.

[54] ILOG Inc., *ILOG CPLEX 7.5, User Manual*, 2001.

[55] Johnson, E. L., Nemhauser, G. L., and Savelsbergh, M. W. P., "Progress in linear programming-based algorithms for integer programming: an exposition," *INFORMS Journal on Computing*, vol. 12, pp. 2–23, 2000.

[56] Kall, P., *Stochastic Linear Programming*. Springer Verlag, Berlin, 1976.

[57] Karmarkar, N., "A new polynomial time algorithm for linear programming," *Combinatorica*, vol. 4, pp. 375–395, 1984.

[58] Khachiyan, L. G., "A polynomial algorithm in linear programming," *Soviet Mathematics Doklady*, vol. 220, pp. 191–194, 1979.

[59] Klein Haneveld, W. K. and van der Vlerk, M. H., "On the expected value function of a simple integer recourse problem with random technology matrix," *Journal of Computational and Applied Mathematics*, vol. 56, pp. 45–53, 1994.

[60] Klein Haneveld, W. K. and van der Vlerk, M. H., "Stochastic integer programming: general models and algorithms," *Annal of Operations Research*, vol. 85, pp. 39–57, 1999.

[61] Krarup, J. and Bilde, O., "Plant location, set covering and economic lot size: An $o(mn)$ algorithm for structured problems.," in *Optimierung bei Graphentheoretischen und ganzzahligen Problemen, International Series of Numerical Mathematics* (Collatz, L., ed.), vol. 36, pp. 155–180, Birkhauser-Verlag, 1977.

[62] Land, A. H. and Doig, A. G., "An automatic method for solving discrete programming problems," *Econometrica*, vol. 28, pp. 493–520, 1960.

[63] Laporte, G. and Louveaux, F. V., "The integer L-shaped method for stochastic integer programs with complete recourse," *Operations Research Letters*, vol. 13, pp. 133–142, 1993.

[64] Laporte, G., Louveaux, F. V., and van Hamme, L., "An integer L-shaped algorithm for the capacitated vehicle routing problem with stochastic demands," *Operations Research*, vol. 50, pp. 415–423, 2002.

[65] Leung, J. M. Y., Magnanti, T. L., and Vachani, R., "Facets and algorithms for capacitated lot-sizing," *Mathematical Programming*, vol. 45, pp. 331–359, 1989.

[66] Linderoth, J. T. and Wright, S. J., "Decomposition algorithms for stochastic programming on a computational grid," *Computational Optimization and Applications*, vol. 2–3, pp. 207–250, 2003.

[67] Loparic, M., Marchand, H., and Wolsey, L. A., "Dynamic knapsack sets and capacitated lot-sizing," *Mathematical Programming*, vol. 95, pp. 53–69, 2003.

[68] Loparic, M., Pochet, Y., and Wolsey, L. A., "The uncapacitated lot-sizing problem with sales and safety stocks," *Mathematical Programming*, vol. 89, pp. 487–504, 2001.

[69] Louveaux, F. and van der Vlerk, M. H., "Stochastic programming with simple integer recourse," *Mathematical Programming*, vol. 61, pp. 301–325, 1993.

[70] Louveaux, F. V., "Multi-stage stochastic programs with block-separable recourse," *Mathematical Programming Study*, vol. 28, pp. 48–62, 1986.

[71] Lulli, G. and Sen, S., "A branch-and-price algorithm for multi-stage stochastic integer programming with application to stochastic batch-sizing problems," 2002. Working paper, Department of Systems and Industrial Engineering, University of Arizona, Tucson, AZ, 2003.

[72] Lustig, I. J., Mulvey, J. M., and Carpenter, T. J., "Formulating two-stage stochastic programs for interior point methods," *Operations Research*, vol. 39, pp. 757–770, 1991.

[73] Marchand, M. and Wolsey, L. A., "The $0-1$ knapsack problem with a single continuous variable," *Mathematical Programming*, vol. 85, pp. 15–33, 1990.

[74] Marti, K., "Solving stochastic linear programs by semi-stochastic approximation algorithms," in *Recent Results in Stochastic Programming, Lecture Notes in Economics and Mathematical Systems 179* (Kall, P. and Prekopa, A., eds.), pp. 191–213, Springer Verlag, Berlin, 1980.

[75] Miller, A. J., *Polyhedral Approaches to Capacitated Lot-Sizing Problems*. PhD thesis, Georgia Institute of Technology, 1999.

[76] Nemhauser, G. L. and Wolsey, L. A., *Integer and Combinatorial Optimization*. Wiley, New York, 1988.

[77] Nemhauser, G. L. and Wolsey, L. A., "A recursive procedure for generating all cuts for 0-1 mixed integer programs," *Mathematical Programming*, vol. 46, pp. 379–390, 1990.

[78] Nowak, M. P. and Romisch, W., "Stochastic Lagrangian relaxation applied to power scheduling in a hydro-thermal system under uncertainty," *Annals of Operations Research*, vol. 100, pp. 251–272, 2000.

[79] Pereira, M. V. and Pinto, L. M. V. G., "Stochastic dual dynamic programming," Technical Note DEE-PUC/RJ, Catholic University of Rio de Janeiro, Rio de Janeiro, Brazil, 1989.

[80] Pflug, G. C., "Stepsize rules, stopping times and their implementations in stochastic quasigradient algorithms," in *Numerical Techniques for Stochastic Optimization* (Ermoliev, Y. and Wets, R. J.-B., eds.), pp. 353–372, Springer Verlag, Berlin, 1988.

[81] Pochet, Y., "Valid inequalities and separation for capacitated economic lot sizing," *Operations Research Letters*, vol. 7, pp. 109–115, 1988.

[82] Pochet, Y. and Wolsey, L. A., "Lot-sizing with constant batches: Formulation and valid inequalities," *Mathematics of Operations Research*, vol. 18, pp. 767–785, 1993.

[83] van Hoesel, C. P. M., Wagelmans, A. P. M., and Wolsey, L. A., "Polyhedral characterization of the economic lot-sizing problem with start-up costs," *SIAM Journal on Discrete Mathematics*, vol. 7, pp. 141–151, 1994.

[84] van Slyke, R. M. and Wets, R. J.-B., "Programming under uncertainty and stochastic optimal control," *SIAM Journal on Applied Mathematics*, vol. 17, pp. 638–663, 1969.

[85] Römisch, W. and Schultz, R., "Multistage stochastic integer programs: an introduction," in *Online Optimization of Large Scale Systems* (Grötschel, M., Krumke, S. O., and Rambau, J., eds.), pp. 581–600, Springer Verlag, Berlin, 2001.

[86] Ruszczyński, A., "A regularized decomposition method for minimizing a sum of polyhedral functions," *Mathematical Programming*, vol. 35, pp. 309–333, 1986.

[87] Ruszczyński, A., "A linearization method for nonsmooth stochastic programming problems," *Mathematics of Operations Research*, vol. 12, pp. 32–49, 1987.

[88] RUSZCZYŃSKI, A. and SHAPIRO, A., eds., *Stochastic Programming. Handbooks in Operations Research and Management Science*, vol. 10. Elsevier Science B. V., 2003.

[89] SCHRIJVER, A., *Theory of Linear and Integer Programming.* John Wiley and Sons, 1987.

[90] SCHULTZ, R., STOUGIE, L., and van der VLERK, M. H., "Solving stochastic programs with integer recourse by enumeration: A framework using Grobner basis reductions," *Mathematical Programming*, vol. 83, pp. 229–252, 1998.

[91] SEN, S. and SHERALI, H. D., "Decomposition with branch-and-cut approaches for two-stage stochastic mixed-integer programming," Technical Report, Department of ISE, University of Arizona, 2004.

[92] TAKRITI, S., BIRGE, J. R., and LONG, E., "A stochastic model for the unit commitment problem," *IEEE Transactions on Power Systems*, vol. 11, pp. 1497–1508, 1996.

[93] WAGELMANS, A., van HOESEL, A., and KOLEN, A., "Economic lot sizing: An $\o(n \log n)$ algorithm that runs in linear time in the Wagner–Whitin case," *Operations Research*, vol. 40, pp. 145–156, 1992.

[94] WAGNER, H. M. and WHITIN, T. M., "Dynamic version of the economic lot size model," *Management Science*, vol. 5, pp. 89–96, 1958.

[95] WETS, R. J.-B., "Programming under uncertainty: The complete problem," *Zeitschrift für Wahrscheinlichkeit und verwandschaftliche Gebiete*, vol. 4, pp. 316–339, 1966.

[96] WETS, R. J.-B., "Stochastic programs with fixed recourse: The equivalent deterministic program," *SIAM Review*, vol. 16, pp. 309–339, 1974.

[97] WETS, R. J.-B., "Solving stochastic programs with simple recourse," *Stochastics*, vol. 10, pp. 219–242, 1983.

[98] WETS, R. J.-B., "On parallel processors design for solving stochastic programs," *Proceedings of the 6th Mathematical Programming Symposium*, pp. 13–36, 1985.

[99] WITTROCK, R. J., "Advances in a nested decomposition algorithm for solving staircase linear programs," Technical Report SOL 83–2, Stanford University, Stanford, CA, 1983.

[100] WOLSEY, L. A., *Integer Programming.* Wiley, New York, 1998.

[101] ZENIOS, S. A., "Massively parallel algorithms for financial modelling under uncertainty," Technical Report, Decision Sciences Department, The Wharton School, University of Pennsylvania, Philadelphia, PA, 1990.

VITA

Yongpei Guan was born in Jiangsu Province, China in 1975. After graduating from high school in 1994, he studied at the Shanghai Jiao Tong University and obtained dual Bachelor's degrees in Mechanical Engineering and Economic Decision Science in 1998. In that summer, he went to the Department of Industrial Engineering and Engineering Management at the Hong Kong University of Science and Technology for graduate studies and he obtained a Master of Philosophy degree in 2001. Then, he moved to Atlanta, Georgia to start his doctoral studies in the School of Industrial and Systems Engineering at the Georgia Institute of Technology.